

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Boris Turk

SPLETNA STORITEV ZA ISKANJE PODATKOV O EKOLOŠKIH KMETIJAH
DIPLOMSKO DELO
UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

Mentor: prof.dr. Igor Kononenko
Ljubljana, 2014

Številka naloge:

Datum: 20.09.2014



Univerza v Ljubljani, Fakulteta za računalništvo in informatiko potrjuje naslednjo nalogo:

Kandidat: Boris Turk

Naslov: **SPLETNA STORITEV
ZA ISKANJE PODATKOV O EKOLOŠKIH KMETIJAH
WEB SERVICE
FOR SEARCHING THE DATA ABOUT ORGANIC FARMS**

Vrsta naloge: Diplomsko delo univerzitetnega študija prve stopnje

Tematika naloge:

Naloga je razviti in implementirati spletno **aplikacijo za iskanje ponudbe ekoloških pridelovalcev** živil v Sloveniji. Spletna aplikacija naj bo sestavljena iz spletne strani z iskalnikom, formami, zemljevidom in predstavitvenimi stranmi uporabnikov, ki bo v bodočnosti lahko razširjena tudi za uporabo na pametnih telefonih. Opišite tudi nekaj alternativnih rešitev na spletu in prednosti izdelane aplikacije pred njimi. Opišite uporabljen tehnologijo, razvoj, orodja, podatkovno bazo, spletne storitve, ter lastnosti kode in načine testiranja aplikacije. Opišite ločeno tudi iskalnik in njegove filtre, podajte primere rezultatov iskanj s povezavami na lokacije ponudnikov in s povezavami na predstavitvene strani ponudnikov ekoloških živil.

Mentor:
prof. dr. Igor Kononenko

Dekan:
prof. dr. Nikolaj Zimic

Rezultati diplomskega dela so intelektualna lastnina avtorja. Za objavljane ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani **Boris Turk**,
z vpisno številko **630 20 169**,
sem avtor diplomskega dela z naslovom
Aplikacija za iskanje podatkov o ekoloških kmetijah

S svojim podpisom zagotavljam, da

- ✓ sem delo izdelal samostojno pod mentorstvom prof. dr. Igorja Kononenka,
- ✓ slednje ne vsebuje materiala, ki bi ga kdorkoli predhodno že objavil ali oddal v obravnavo za pridobitev naziva na univerzi ali drugem visokošolskem zavodu, razen v primerih, kjer so navedeni viri
- ✓ so elektronska oblika dela, naslov (slov.,angl.), povzetek (slov.,angl.) ter ključne besede (slov.,angl.) identični s tiskano obliko in
- ✓ soglašam z javno objavo elektronske oblike dela v zbirki »Dela FRI«

V Novem mestu, dne 25. september 2014

Podpis avtorja:

Zahvala

Zahvaljujem se vsem, ki so kakorkoli pripomogli k uspešni izvedbi diplomskega dela, k mojemu pridobivanju informacij s strani ekologije, in tistim, ki so te informacije nudili.

Zahvaljujem se tudi svojemu prvemu vrtu, na katerem sem v času izdelave diplomske naloge v praksi preizkušal rešitve miniaturnega ekološkega kmetovanja. Vrt je bogato obrodil in nikoli prej nisem vedel, da lahko vrt v izmeri sedemnajstih kvadratnih metrov predstavlja 1/3 mojega dnevnega jedilnika.

Posebna zahvala gre:

- Tistim, ki so se z menoj pogovarjali o ekoloških problemih. Tu bi težko izpostavil bilokoga, saj bi najverjetneje koga izpustil.
- Vsem, ki so me več let osveščali o pomembnosti ekološkega načina pridelave.
- Vsem, ki so z menoj diskutirali o spletni aplikaciji in podali mnenja o njej. To so bili: Tina Pahor, Jure Rajer, Toni Turk, Jasna Turk, Darinka Turk, Anže Žnidaršič, Katja Šercelj, Boža Šercelj, Oto Hegediš, mentor Igor Kononenko, Ermin Kotorič, Maša Kastelic, Sandra Borse. Vsa mnenja so bila upoštevana in so pripomogla k njeni boljši izvedbi.

Zahvala gre tudi Nives Adamič Kričaj, koordinatorki kontrole ekološkega kmetijstva in Maji Žibert, zaposleni na Ministrstvu za kmetijstvo. Nudili sta mi uporabne informacije o certifikatih in stanju ekološkega kmetovanja v Sloveniji.

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Motivacija	2
1.2	Cilji diplomskega dela	2
1.2.1	Cilji, ki se nanašajo na prehransko varnost, gospodarstvo in zaposlenost	3
1.2.2	Ekonomsko usmerjeni cilji	4
1.2.3	Ekološko usmerjeni cilji	5
1.3	Pregled vsebine	6
2	Metodologija	7
2.1	Primerjava aplikacije z obstoječimi rešitvami na spletu	8
2.1.1	Iskalniki	8
2.1.2	Statični sezname podatkov	10
2.1.3	Prednosti aplikacije	11
2.2	Učinkovitost aplikacije	12
2.2.1	Uporabnost - koristi aplikacije	12
2.2.2	Dostopnost aplikacije	13
2.2.3	Prepoznavnost in všečnost aplikacije	13
2.2.4	Optimizacija aplikacije za spletne iskalnike ter širjenje glasu	14
3	Razvoj aplikacije	17
3.1	Tehnologije, razvojna in programska orodja	19
3.2	Nastavitev jezika	20
3.3	Dostop do podatkov	21
3.3.1	Dostop do podatkov spletne strani s strani brskalnika	21
3.3.2	Dostop do podatkov podatkovne baze s strani skripte PHP	21
3.3.3	Vsebina datoteke index.php	23
3.3.4	Podatkovna baza	28
3.4	Spletna stran	44
3.4.1	Uporaba iskalnika	44
3.4.2	Delovanje iskalnika - spisek ekoloških kmetij glede na izbrane filtre	45
3.4.3	Prikazani podatki v rezultatih iskanj	53
3.4.4	Zemljevid ekoloških kmetij	54
3.4.5	Prijava in registracija	56
3.4.6	Predstavitvene strani in administrativna stran za uporabnike	59
3.4.7	Uporabljene knjižnice in moduli	65
3.5	Preizkušanje aplikacije	66
3.6	Primer skripte PHP za primerjanje podatkov v bazi s tujimi podatki	68
4	Sklep in nadaljnje delo	70
4.1	Sklepne ugotovitve	70
4.2	Nadaljnji razvoj	70
	Literatura	71

Kazalo dodatkov

Dodatki	73
1 Kaj je ekološka pridelava?	73
2 Kmetijstvo v Sloveniji	73
2.1 Analiza stanja ekološkega kmetovanja	74
2.2 Prehranska varnost in samooskrba	75
3 Zakaj število kmetijskih površin upada	77
3.1 Kako zagotoviti prehransko varnost?	79
3.2 Ali lahko povečamo kmetijske površine?	80
4 Prednosti ekološkega kmetovanja	81
Literatura dodatkov	83

Povzetek

Izdelek diplomskega dela je spletna aplikacija z iskalnikom, ki obiskovalcu omogoča iskanje podatkov o ponudbi ekoloških pridelovalcev živil v Sloveniji. Pri tem lahko uporabi filtre, s katerimi omeji rezultate iskanj, ki se asinhrono prikažejo na isti strani.

Filter »Iskalna fraza« je splošen in omogoča obiskovalcu vpis besed, katere aplikacija išče po več tabelah znotraj baze. Iskati je mogoče po več besedah hkrati, pri čemer se lahko uporablja operatorja IN (znak &) in ALI (vejica). Poleg splošnega so na voljo še filtri z vnaprej podanimi opcijami: blagovna znamka, regija, poštni okraj, status kmetije, ponudba kmetije, posebnosti.

Rezultat iskanja je spisek ekoloških kmetij v obliki tabele, v kateri vsaka vrstica predstavlja eno kmetijo. Stolpci tabele so: regija in polni naslov, kontaktna oseba, naziv in podrobnosti, kontakt, združenje, čas in kraj prodaje, pridelki in živila. V rezultatih so zastopane tudi povezave do predstavitev strani pridelovalcev, njihovih zunanjih spletnih strani in lokacij na zemljevidu Google maps. S slednjim je aplikacija povezana še na en način – ponuja zemljevid z lokacijami pridelovalcev ekoloških živil, kategoriziranih v skupine glede na certifikate.

Poleg iskalnika vsebuje spletna aplikacija tudi predstavitevne strani pridelovalcev ekoloških živil. Pridelovalcu omogoča prijavo ali registracijo, v kolikor njegovega e-poštnega naslova še ni v naši bazi. V administrativnem delu lahko posodablja nekatere svoje podatke in ustvari formatiran opis kmetije.

Uporabljene tehnologije: PHP, jQuery, MySQL, Java Script, Ajax, HTML, CSS.

V diplomski nalogi je zajet opis problema šibke samooskrbe in prehrambene varnosti v Sloveniji, splošnega redčenja živalskih vrst na račun uporabe fitofarmaceutskih proizvodov. Predstavljeno je, kako ekološko kmetijstvo rešuje okoljske in gospodarstvene probleme,...

Ključne besede: spletna aplikacija, spletni iskalnik, ekološke kmetije, ponudba živil ekoloških pridelovalcev, XMLHttpRequest

Abstract

The thesis represents the solution how to search organic farms food offer data in Slovenia. This solution is implemented as a web service with a search mechanism. User can use filters to narrow search results which represents on the screen asynchronously.

There are two types of filters. Filter »Search phrase« allows user to limit search results by entering their own words. There the user can use operators AND (character &) and OR (comma). Other filters are static - options are given in advance. These filters are: brand name, district, ward, farm status, farm offer, specialities.

Searching results is a list of organic farms, represented as a table. Each table row represents the data about one farm. Table columns: district, full address, contact person, farm title, details, contact, organisation, time and place to sell, harvest and food. The results also contain links to farms presentation pages, offsite web pages, their Google maps locations. The application is linked to Google maps in one another way – it offers a map with organic farms food producers, categorized into groups, based on certificate.

Web application includes presentation pages for each organic food producer. Producers are able to login into web application or register, in case their e-mail is not stored in our data base. When logged in, the user is redirected to administration page, where he or she can update some of the data and create his/her own HTML presentation .

Used technology: PHP, jQuery, MySQL, Java Script, Ajax, HTML, CSS.

The thesis also brings out the problem of weak food self-supply and alimental safety in Slovenia, and general reduction of animal species, which is the consequence of using fito pharmaceutical means. It is explained how organic farms solve environmental and economical problems.

Key words: web application, web service, search engine, organic farms, organic producer food offer, XMLHttpRequest

1 Uvod

Diplomsko delo je iz računalniške stroke in predstavlja spletno aplikacijo za iskanje po ponudbi ekoloških pridelovalcev živil. Zastavljeno je na način, da ob tem širi zavest o pomembnosti ekološkega načina pridelovanja živil, povečevanja števila ekoloških kmetij, povečevanja prehranske samooskrbe in prehranske varnosti. Različne kmetijske in okoljske institucije v Sloveniji pričajo, da so ta področja v Sloveniji kritično problematična, kar je eden izmed glavnih razlogov, zakaj je bila aplikacija izdelana. Zato je s to tematiko prežetih pet strani diplomske naloge. Bralec je preko poglavij postopoma voden od osnovnih konceptov do razdelanih analiz, kjer se preko podatkov, pridobljenih s Statističnega urada in drugih virov, lahko sam prepriča o realnosti opisanih problemov in učinkoviti predlaganih rešitev.

Pri raziskovanju je bilo ugotovljeno da:

- Imamo resne probleme na račun monokultur, ki uničujejo raznovrstnost, izčrpavajo hranilne snovi v tleh, povečujejo erozijo tal.
- So problemi z zdravjem ljudi v večji meri povezani z načinom pridelave in priprave hrane.
- Na račun konvencionalnega kmetijstva, kjer se uporabljajo fitofarmacevtska sredstva, vsako leto umre ogromno čebeljih družin, ki pa niso pomemben člen le za človeka, pač pa tudi za druga živa bitja, saj oprašujejo vrtnine, rože, oprašijo 70 do 80 odstotkov cvetov sadnega drevja,... Število živalskih in rastlinskih vrst (ne zgolj število pripadnikov vrste) upada (dobesedno) nekaj tisočkrat hitreje, kot v več tisočletni zgodovini.
- Povečanje števila ekoloških kmetij situacijo bistveno izboljša.

Med tem, ko nekateri kmetovalci škropijo polja s fitofarmacevtskimi pripravki, da bi imeli večji dobiček, ustvarjajo neznosne razmere v ekosistemu (polja izgledajo kot »puščave«, na katerih raste ena sama rastlinska vrsta). Po drugi strani pa nekateri ekološki permakulturniki^A ustvarjajo rodovitne, pestre oaze v puščavah, na katerih po prepričanjih nekaterih strokovnjakov ne more rasti nič.

^A Permakulturnik Geoff Lawton na strani <http://www.geofflawton.com/> nudi brezplačne videoposnetke, v katerih opisuje različne metode permakulture skozi prakso.

1.1 Motivacija

Slika 1: Greda velikosti 17 m²



To je pot k večji ekološki samooskrbi s praktičnega vidika. Greda (slika 1) je del ekološkega vrta v velikosti 17 m². Zelenjava raste brez uporabe fitofarmaceutskih pripravkov, na rodovitni zemlji z ugodnim pH 6,8. Stroški izdelave vrta (stroški prevoza strohnelih hlodov, listja) in nakupa semen, sadik ter zaščitne folije: 38 €. Z vrta se vsak dan znosi cca 0,5 kg užitnih rastlin, vendar rast prehiteva porabo za 2x. Greda je med drugim polna rukole in radiča, ki na trgu staneta nad 5 €/kg. Minimalna neto plača avgusta 2014: 789,15 €.^B

V enem kilogramu takšnih rastlin je do 500 kCal, obilo mineralov in vitaminov. Če bi na tak način pridelali še ostala živila, bi se lahko ena oseba prehransko preživela na 100 m² površine. V dodatku lahko preberemo, da pridelava na konvencionalni način ne daje toliko donosa. Ekološka pridelava lahko nudi dovolj velik donos, je pa potreben sistematični pristop, ustrezna umestitev rastlin na ustrezno mesto, kompostiranje, zastiranje, kolobarjenje,...

Zamislite si scenarij, da potujete po Sloveniji in vas želodec opozori, da bo potrebno zaužiti nekaj hrane. Namesto v bližnjo trgovino bi se lahko napotili k najbližjemu ekološkemu pridelovalcu in prodajalcu živil po svežo lokalno hrano, brez ostankov nepotrebnih pesticidov.

Spletna aplikacija omogoča, da preko spleta (tudi preko pametnih telefonov) poiščemo ekološke kmetije. Iskalnik z naprednimi filtri vrača rezultate iz baze več tisoč pridelovalcev ekoloških živil v Sloveniji. Rezultati iskanj se pojavijo na isti strani in prikazujejo nazive kmetij, kontaktne osebe, kontaktne podatke kmetije, seznam živil v ponudbi, čas in kraj prodaje, povezave na strani s podrobnostmi, avtorske strani, certifikate, zemljevide,...

1.2 Cilji diplomskega dela

Cilji diplomskega dela so naravnani k razširjanju zavesti o pomembnosti ekološkega načina kmetovanja, povečanju prehranske samooskrbe in ob enem zadostitev pogojev za doseganje ustrezne prehranske varnosti. Na poti k temu naletimo na precej ovir, katere pa je možno premostiti z boljšo organizacijo, sistematičnim pristopom, povezanostjo.

Na račun aplikacije, ki je izdelana v okviru diplomskega dela, se bo lahko krepila vez med potrošnikom in pridelovalcem ekoloških živil, povečal krog zanimanja za ekološko hrano,

^B Podatke o minimalni plači objavlja Ministrstvo za finance RS na strani www.durs.gov.si

omogočil lažji/hitrejši dostop do ekoloških živil in pridelovalcev, ki bi želeli deliti svoje izkušnje in informacije o ekološkem načinu pridelave hrane. Ideja je tudi, da se lahko na račun tega zmanjšajo cene ekoloških živil. Aplikacija mora zato ustrezati nekaterim kriterijem, ki se nanašajo na uporabnost, ažurnost, dostopnost, prepoznavnost, vsečnost, optimizacijo za spletne iskalnike.

Spletna aplikacija ponuja rešitev za izboljšanje organizacije ekološkega trga. Organiziranost pripomore k hitrejšem iskanju živil, večjem zaupanju in zadovoljstvu. To s seboj prinese večje povpraševanje in s tem boljše možnosti za večjo ponudbo in razvoj ekoloških kmetij. Več eko kmetij → večja skrb za trajnostni razvoj, večja biotska raznovrstnost, večja samooskrba → boljše gospodarstvo → manjša brezposelnost. Seveda obstajajo tudi pomembni zdravstveni razlogi, katere pa bi bilo zaradi kompleksnosti treba obravnavati ločeno, zato v diplomskem delu niso posebej razdelani.

1.2.1 Cilji, ki se nanašajo na prehransko varnost, gospodarstvo in zaposlenost

- Stele in Žaucer [1] v publikaciji z leta 2013 pišeta o kmetijstvu doma in drugje po EU: *»Čeprav kmetijstvo v EU-27 prispeva le 1,2 % k celotnemu BDP, je ena najpomembnejših gospodarskih panog. Kmetijstvo namreč poleg svoje najpomembnejše naloge, to je oskrba prebivalstva s kakovostno hrano, skrbi tudi za ohranitev kulturne krajine, pozitivno učinkuje na razvoj podeželja, na ohranjanje poseljenosti, še posebej na demografsko ogroženih območjih, ter ustvarja tudi delovna mesta.«*
- Na spletnih straneh SURS [2] preberemo, da ne upada zgolj št. kmetijskih gospodarstev, ampak hkrati tudi delovna sila. Pomembnejši podatek je npr. da je leta 2000 družinska delovna sila (drugi člani gospodinjstva brez gospodarja) štela 166 192 oseb, leta 2010 pa le še 130 303. Cilj aplikacije je pripomoči k temu, da bi se v (ekoloških) kmetijskih gospodarstvih povečala delovna sila.
- Na straneh Ministrstva za kmetijstvo, gozdarstvo in prehrano RS je dostopna publikacija [3], v kateri piše, da je EU 17. junija 2010 sprejela strategijo Evropa 2020, to je novo strategijo za rast in delovna mesta. Izvleček: *»Strategijo Evropa 2020 bodo morale podpreti vse politike, zato bo ta strategija okvir tudi za bodočo SKP. Kmetijstvo namreč prek svojih funkcij prispeva in bo tudi v prihodnje pomembno prispevalo k ohranjanju rasti in delovnih mest na evropski ravni in s tem k prioritetam Strategije Evropa 2020. Kmetijstvo in proizvodnja hrane imata pomembno vlogo pri trajnostni rasti. Temeljna naloga preskrbe z varno hrano, proizvedeno na trajnostni način, je eden od najpomembnejših izzivov, zato je vloga kmetijstva ponovno postala strateškega pomena« ... »V okviru prioritete vključujoče rasti predstavljata kmetijstvo in proizvodnja hrane bistven del podeželskega gospodarstva, saj prispevata k ohranjanju prebivalstva in gospodarske dejavnosti na podeželju, tako preko osnovne kot številnih komplementarnih, dopolnilnih dejavnosti kmetijstva...«*. Rešitev, ki jo predstavlja diplomska naloga, bo pripomogla k uresničevanju tega cilja, saj bo preko spletnega iskalnika potrošnika približala pridelovalcu ekoloških živil in bo lahko prispevala k izboljšanju prehranske samooskrbe znotraj države.

1.2.2 Ekonomsko usmerjeni cilji

- Omogočiti ekološkim pridelovalcem brezplačno promoviranje pridelanih živil.
- Stimulirati potrošnike k nakupu lokalnih ekoloških živil; kmetijskim gospodarstvom izboljšati možnosti za prodajo neposredno potrošnikom in s tem ustvarjati boljše pogoje za prodajo teh živil po nižji ceni (ni birokracije, ni marže, ni vmesnih členov, ki dražijo produkte).
- Omogočiti boljšo organizacijo manjšim trgovinam, ki ne najdejo, oziroma imajo težave pri iskanju dobaviteljev lokalnih ekoloških živil. Podobno velja za vrtce in šole.
- Evropska unija je Sloveniji že večkrat naklonila sredstva, ki pa so bila po odločitvah vlade večkrat namesto v trajnostni razvoj in družbeno koristne razvojne programe, namenjena parcialnim interesom vplivnih lobijev. Ker nam je »skleda podarjenega zdravega obroka« tako večkrat odnešena, je dobro imeti alternativne rešitve, pri katerih nismo odvisni od tretjih členov.
- Povečati povpraševanje po prodaji živil neposredno potrošnikom. Z vira [1] so pridobljeni strnjeni podatki o trenutnem stanju glede števila in površine kmetijskih zemljišč glede na namen pridelave v splošnem. Večina kmetijskih gospodarstev prodaja preko posrednika, ki za prodajano blago zaračuna maržo, zato je cena blaga po eni strani višja, po drugi strani pa je pridelovalec plačan manj, kot bi bil, če bi lahko več živil prodal neposredno potrošnikom. Odkupne cene živil s strani veletrgovcev (preprodajalcev) so namreč nemalokrat zelo nizke, konkurenca s tujega trga pa jih še dodatno niža.

Namen kmetijske pridelave za leto 2010

	Število kmetijskih gospodarstev	Površina kmetijskih zemljišč v uporabi KZU (ha)
Namen pridelave – skupaj	74646	474432
Izključno za lastno porabo	34628	92563
Pretežno za lastno porabo	9798	35001
Prodaja neposredno potrošnikom	7600	78473
Prodaja preko posrednika	22620	268394
Pri površini kmetijskih zemljišč niso vključeni skupni pašniki (8.221ha skupnih pašnikov). KZU – kmetijska zemljišča v uporabi.		

1.2.3 Ekološko usmerjeni cilji

- Razširjati zavest o pomembnosti eko kmetovanja. Aplikacija nudi bistvene informacije o ekološkem načinu pridelave živil. Ekološko kmetijstvo bistveno prispeva k ohranjanju biotske raznovrstnosti, kulturne krajine in zaščiti naravnih virov za vzdržnejši ekosistem ter boljše počutje ljudi in živali.
- Aplikacija omogoča hiter, pregleden dostop do podatkov o pridelovalcih ekoloških živil in njihovi ponudbi. Cilj je, da lahko več ljudi pride do sezone, ekološke, sveže hrane.
- Glede na to, da je bilo po viru [3k] v letu 2013 v ekološko kontrolo vključenih 3.049 kmetijskih gospodarstev (kar predstavlja 4,1% vseh kmetij v Sloveniji) z 38.664,49 ha kmetijskih zemljišč v uporabi (8,4% od vseh kmetijskih zemljišč v uporabi v letu 2013) se ponudi vprašanje, čemu aplikacija ne ponuja iskanja podatkov o vseh kmetijah po Sloveniji. Odgovorov je kar nekaj in jih bralec lahko razbere iz prilog diplomske naloge. Na kratko povedano, promovira se način, ki je celostno gledano boljši za ekosistem in tudi za gospodarstvo za generacije za nami.

1.3 Pregled vsebine

Drugo poglavje predstavlja, kako je potekalo raziskovanje problemov prehranske varnosti, samooskrbe in ekoloških kmetijskih gospodarstev. Omenjene so nekatere spletne rešitve, na račun katerih se lahko ti problemi zmanjšujejo.

Bolj celostno rešitev predstavlja aplikacija, o kateri govori diplomsko delo. Predstavljene so njene prednosti in slabost, ki je bolj politične narave in za katero upamo, da se bo z dogovori odpravila. Nato je pojasnjeno, kako aplikacija upravičuje svoj smisel, kakšne koristi so od nje, kako jo lahko uporabljamo in kako jo bomo lahko uporabljali po nadgradnji. Preberemo lahko, kaj je bilo storjeno, da bi bila aplikacija učinkovita.

Pri izdelavi so bila poleg programerskih upoštevana tudi teoretična in izkustvena znanja s področja oblikovanja ter optimizacije spletnih strani. Pojasnjeno je, kaj točno se želi z aplikacijo doseči, kateri so uresničljivi cilji in katere napovedi so optimistične.

Tretje poglavje govori o razvoju aplikacije, tehnologiji, programskih orodjih, jeziku, načinu dostopanja do podatkov strani in podatkov iz podatkovne baze, spletnih storitvah, kodi aplikacije. Posamezno so obravnavani iskalnik in njegovi filtri, pregleden spisek rezultatov iskanj, obogaten s povezavami na lokacije ponudnikov, menuji, povezavami na predstavitevne strani ponudnikov ekoloških živil znotraj spletišča in povezavami na zunanje spletne strani.

Prav tako sta posamezno obravnavana zemljevid ekoloških kmetij, pri katerem so kmetije kategorizirane po certifikatih, in forma za prijavo ter registracijo, pri kateri se kot antispam orodje uporablja captcha, gesla in uporabniška imena pa se preverjajo sproti, asinhrono (v ozadju).

Predstavljen je mehanizem, ki omogoča, da ima vsaka kmetija svojo predstavitevno stran. Pojasnjeno je, kako lahko verificirani uporabnik spreminja podatke svoje predstavitvene strani.

Posebej so predstavljene tudi nekatere uporabljene knjižnice in način uporabe.

Nazadnje je opisano, kakšni pristopi in mehanizmi so bili uporabljeni za testiranje aplikacije. Vsa podpoglavja tretjega poglavja so opremljena s slikami in primeri programske kode.

Na koncu si lahko v četrtem poglavju preberemo nekaj planov za razširitev oziroma nadgradnje aplikacije, ki bodo vsekakor v prihodnosti realizirane.

Štirje dodatki so namenjeni predstavitvi ekološke pridelave in problemov kmetijskega gospodarjenja v Sloveniji ter navedbi prednosti ekološkega kmetovanja. Tu so stvari razdelane bolj podrobno, na ogled so razne preglednice in grafi.

2 Metodologija

Kako pripomoči k povečanju števila ekoloških kmetij? Raziskati je bilo potrebno situacijo v politiki, ugotoviti s kakšnimi problemi se soočajo ekološke kmetije, kako funkcionirajo zadeve na trgu. Raziskati je bilo potrebno, kje so zavore, zakaj je ekološka hrana dražja, kako je prišlo do tega, da je bilo januarja 2014 registriranih brezposelnih oseb med aktivnim prebivalstvom več kot 14 %, po drugi strani pa ljudje zapuščajo kmetijska gospodarstva, ki so v bistvu vir hrane in zaslužka. S strani Ministrstva za kmetijstvo in okolje je bilo v tej smeri opaziti precej prizadevanj, tako za povečanje števila kmetij, kot za ekološke načine obdelave kmetij. Bralec lahko to razbere iz navedenih virov in dodatkov k diplomski nalogi. Po drugi strani pa bi bilo s strani vlade pričakovati večjo angažiranost, več programov in subvencij, saj število kmetijskih gospodarstev še vedno upada. Na srečo je na strani ekoloških kmetijskih gospodarstev slika drugačna - v zadnjih letih njihovo število narašča. Ampak s tem problemi za ekološko stran še niso rešeni. Problem moramo rešiti ljudje sami – potrošniki se moramo koordinirati z ekološkimi pridelovalci živil Na slovenskem območju.

Če si, na primer, pozimi zaželimo ekološka jabolka, lahko ugotovimo, da so cene v supermarketu približno 3x dražje od neekološko pridelanih. Kje lahko dobimo cenejša? Neposredno pri pridelovalcu. Na spletu je za laika iskanje ponudnikov ekoloških živil nekoliko (pre)zakomplicirano in časovno (pre)potratno. Eden od načinov reševanja tega problema je uporaba spletne aplikacije za kategorično iskanje po ponudbi ekoloških pridelovalcev živil, ki ima več filtrov, kot jih nudijo obstoječi splošni ali specializirani iskalniki. Diplomsko delo predstavlja izvedbo te rešitve.

Pred začetkom programiranja je bilo potrebno raziskati, komu lahko aplikacija koristi, kaj lahko izboljša, kdo bi jo želel uporabljati, kako bodo ljudje za njo izvedeli, kakšne funkcionalnosti bi bilo vredno dodati. S pomočjo orodja Google keyword tool je bil preiskan trg za ključne besede, ki se nanašajo na iskalnik ekoloških kmetij. Skovane so bile osnovne ideje tega, kako bi lahko ta aplikacija pripomogla k reševanju vseh zgoraj omenjenih problemov. Ideja je bila, da bi lahko potrošnik na katerem koli koncu Slovenije na čim bolj enostaven način našel ponudbo najbližjih ekoloških kmetij, videl njihovo spletno predstavitev, njihovo lokacijo na zemljevidu in jih kontaktiral. Aplikacija omogoča vse to na eni sami strani.

Pred začetkom razvoja aplikacije je bilo potrebno zbrati podatke o ekoloških pridelovalcih. Na spletu je že obstajal preprost iskalnik ekoloških kmetij [5]. Ustvariti je bilo potrebno podatkovno bazo s tabelami, nato pa sprogramirati spletno skripto, ki je z referenčnih podstrani pridobila podatke in jih zapisala v bazo. Kasneje je bilo ugotovljeno, da imamo v Sloveniji več kot 10 x več ekoloških kmetij, kot jih je v tisti bazi. V Sloveniji so tri kontrolne organizacije, ki izvajajo postopek certificiranja ekološkega kmetijstva. Vse tri kontrolne organizacije vodijo evidence preko Ministrstva za kmetijstvo (MKO).

Spletna aplikacija je realizirana kot spletna stran, povezana s spletnim strežnikom in podatkovno bazo. Dostopna je na naslovu <http://ekološke-kmetije.si>.

2.1 Primerjava aplikacije z obstoječimi rešitvami na spletu

Na slovenskem območju so podatki o ekoloških kmetovalcih zelo razkropljeni. Nekakšen red je vzpostavljen preko Ministrstva za kmetijstvo in okolje, vendar je iskanje po podatkih zamudno in zelo nepraktično. Ministrstvo za kmetijstvo in okolje razpolaga s podatki o tem, kje so ekološke kmetije, s čim se ukvarjajo in kaj pridelujejo. Po zakonu pa so nekateri podatki o pridelovalcih zaupni in jih ne smejo posredovati. Treba je torej najti spletišča, na katerih so pridelovalci ekoloških živil objavili podatke po svoji volji, oziroma so dovolili objavo teh podatkov. Nekaj jih najdemo na spletnih straneh ekoloških blagovnih znamk, pod katere se združujejo ekološki kmetovalci.

2.1.1 Iskalniki

Pregledani so bili trije tipi iskalnikov: 2 splošna, 3 iskalniki poslovnih subjektov in 1 iskalnik, specializiran za iskanje ekoloških kmetij.

2.1.1.1 Splošni iskalniki

Vpis iskalne fraze »"ekološka kmetija"« (z navednicami) v spletni iskalnik Google^C vrne 196 zadetkov. Vpis fraze »ekološke kmetije« (brez navednic) pa 221 zadetkov. Pri tem smo upoštevali realne zadetke, do katerih pridemo s klikanjem na podstrani, ne številko, ki podaja oceno števila zadetkov, ki se izpiše na prvi strani. Slednja je večja od nekaj tisoč. Vpis iskalne fraze »ekološka kmetija« v telefonski imenik Slovenije vrne 37 zadetkov.

2.1.1.2 Trije iskalniki poslovnih subjektov

1. Slovenska knjiga^D je avtor poslovnega informatorja Republike Slovenije (PIRS) – spletnega iskalnika večine poslovnih subjektov v Sloveniji, vendar če v njihov iskalnik vpišemo »ekološka kmetija«, dobimo zgolj 24 zadetkov. Za posamezne zadetke lahko preverimo kontaktne podatke, kje se nahajajo (opremljeno z zemljevidom), kategorije njihovih storitev in izdelkov. Poslovni subjekt lahko proti plačilu doda tudi svojo predstavitevno stran.
2. Tsmedia^E je avtor poslovnega imenika – spletnega iskalnika, ki na vpis iskalne fraze »ekološka kmetija« vrne 21 zadetkov. Podatki, ki jih dobimo o ponudnikih, so podobni, kot pri zgornji rešitvi. Predstavitvene strani proti plačilu.
3. SONOS GmbH^F je avtor spletnega iskalnika, ki na vpis iskalne fraze »ekološka kmetija« vrne 31 zadetkov. Na strani zadetkov je za vsako kmetijo podan opis v treh vrsticah, logotip, regija. V podrobnostih pa pridemo do celovitega opisa kmetij. Na voljo so številne slike kmetij in njihovih prodajnih produktov ter zemljevid. Za predstavitev na eni spletni strani kmetija odšteje 130 € letno.

^C Google Inc.: Spletni iskalnik Google: www.google.com

^D Slovenska knjiga d.o.o: Poslovni informatorj Republike Slovenije (PIRS): <http://www.pirs.si>

^E TSmedia, medijske vsebine in storitve, d.o.o.: Poslovni imenik: <http://www.bizi.si/>

^F SONOS GmbH: Spletni iskalnik Info Slovenija. <http://www.info-slovenija.info/>

Noben od navedenih iskalnikov ne omogoča nekega pametnega iskanja po ekoloških živilih. Če na primer v iskalnik PIRS vpišemo besedo »jabolka«, bo izpisanih nekaj zadetkov, vendar ne bomo vedeli ali gre za ekološka.

2.1.1.3 Iskalniki ekoloških kmetij

Na strani MKO [4] je navedeno: »Ministrstvo za kmetijstvo in okolje je na svoji spletni strani objavilo spletni iskalnik ekoloških certifikatov, ki so jih pridelovalcem in predelovalcem ekoloških proizvodov ter proizvodov iz obdobja preusmeritve, izdale organizacije za kontrolo in certificiranje, ki so s strani Ministrstva za kmetijstvo in okolje, pooblašene za izvajanje kontrole ekološke pridelave in predelave. ... Namen iskalnika je, da potrošniki lahko preverijo verodostojnost certifikata, ki so jih izdale te organizacije za kontrolo in certificiranje, za posamezne ekološke proizvode in za proizvode iz obdobja preusmeritve.«

Njihov spletni iskalnik ekoloških certifikatov je žal zelo okrnjen in ni namenjen promociji, pač pa verifikaciji že poznane ekološkega kmetovalca. Išče se lahko zgolj po številki certifikata ali priimku ali nazivu pridelovalca. Če vnesemo zraven še ime pridelovalca, nam ne bo iskalnik našel ničesar. Išče lahko torej le, če že poznamo pridelovalca. Na tak način so rešili problem, kako potrošniku omogočiti, da preveri ali kmetovalec slučajno ne razpolaga z lažnim certifikatom, hkrati pa ne delujejo v nasprotju z zakonom, ki prepoveduje posredovanje zaupnih kontaktnih podatkov.

Slika 2 - zgled rezultatov iskanja na iskalniku www.eko-kmetije.info

Rezultati iskanja

Iskalni kriteriji

Ključna beseda jabolka

Regija Status kmetije Ponudba kmetije Biodar

Ime kmetije	Pošta	Kraj	Telefon	Status
Ambrožič Lidija	1385	Nova Vas	01/709 37 92	E
Arnež Jože	4205	Predvdor	04/2551458	E
Ažnoh Anton	2380	Slovenj Gradec	02/8844862	E
Baligač Jožef	9231	Beltinci		E
Bogataj Dušan	5281	Spodnja Idrija	05/377-90-96	E
Bogovič Rajko	8253	Artiče		E
Bračič Boris	3273	Jurklošter	03/491-11-86	E
Bračko Ivan	2000	Maribor	02/252 66 70	E
Bračun Glušič Nataša	3000	Celje	03/580 82 00	E
Bravec Edvard	1414	Podkum	03/567 63 11	E
Bregar Pavla	1301	Krka	01/780 20 38	E
Brezočnik Vojko	2344	Lovrenc na Pohorju	02/675-00-11	E
Brezovnik Peter	3332	Rečica ob Savinji	03/583 52 99	E
Cankar Slavka	1360	Vrhnika	01/750-53-22	E
Cesar Danica	3224	Dobje pri Planini	03/579 63 83	E
Cirar Avguštin	1270	Litija	01/898 91 28	E
Cveček Dušan	2208	Pohorje	02/603 31 71	E
Čečko Martina	8294	Boštanj	07/814-29-47	E
Čermelič Zvonko	8253	Artiče	07/495 66 46	E
Čevka Dominik	1242	Stahovica	01/832 51 69	E
Čič Tone	1262	Dol pri Ljubljani	01/563 82 60	E
Črv Viktorija	5283	Slap ob Idrijci	05/380 95 46	E
Demšar Peter in Brigita	4226	Žiri	04/519 14 13	E
Dravčbaher Anton ml.	2367	Vuzenica	02/876 47 43	E
Dušej Aleksander	2000	Maribor	059 1227 22	E

Zadetki 1 - 25 od skupaj 180

< 1 2 3 4 5 6 7 8 >



Nazaj na iskanje

Inštitut za kontrolo in certifikacijo v kmetijstvu in gozdarstvu [5] je pripravil spletni vodnik po ekoloških kmetijah. Nagnjeni so k promociji slovenske blagovne znamke Biodar. Njihov iskalnik omogoča iskanje po ključnih besedah, regiji, statusu kmetije, specifični ponudbi kmetije, znamki Biodar. V rezultatih iskanj pridobimo spisek kmetij na več podstraneh. Za vsako kmetijo so na voljo še podrobnosti (kontaktni podatki, čas in kraj prodaje, opis poti do kmetije, pridelki v ponudbi). Primer izpisa podatkov njihovega iskalnika prikazuje slika 2.

2.1.2 Statični sezname podatkov

Sinergise^G je avtor spletišča, na katerem se nahajajo zemljevidi za več različnih kategorij: restavracije, lokali, razne šole, bolnišnice,... Med njimi je tudi zemljevid ekoloških kmetij. S klikom na posamezno značilko na zemljevidu kmetij se prikažejo podrobnosti o kmetiji, ki so podobne podrobnostim, ki jih najdemo v rezultatih iskanj zgoraj omenjenega iskalnika z Inštituta za kontrolo in certifikacijo.

Na spletišču zavoda Demeter^H lahko najdemo podatke o biodinamičnem kmetovanju in nekaj predstavitvenih strani kmetij, ki so usmerjene v takšen način kmetovanja. Za leto 2014 so za svojo blagovno znamko pripravili tudi seznam prijavljenih kmetij za kontrolo. Na seznamu najdemo kontaktne podatke 12-ih ekoloških kmetij in nekaj kmetij, ki so še v kontroli za pridobitev certifikata. Njihov režim za pridobitev certifikata je precej strožji, saj zahtevajo več znanja o zakonitostih naravnega gibanja Zemlje in letnih časov.

Zveza Biodar^I objavlja seznam uporabnikov znamke Biodar, v katerem najdemo približno 100 nazivov pridelovalcev in njihovih kontrolnih števil. Na podstrane lahko pridobimo še njihove naslove, kontaktne podatke in opis ponudbe.

Zveza ZZEKS^J je zveza več regionalnih združenj slovenskih ekoloških kmetov. Pripravili so celoten seznam ekoloških kmetij (septembra 2014 seznam šteje 2682 kmetij), v katerem je naveden naziv, naslov, pošta in kraj ter regija. Klik na posamezen naslov odpre lokacijo na Google zemljevidu.

Animata^K je pripravila zanimiv seznam ekoloških trgovin in kmetij v Sloveniji, kjer lahko izpis omejimo glede na omrežno skupino, katero izberemo s klikom na del slike Slovenije. V podatkih so navedeni naziv, naslov, telefon, mobilni telefon, e-pošta. Na podstrane (podrobnosti) je še lokacija kmetije na najdi.si zemljevidu.

IRT^L je pripravil seznam ekoloških kmetij, ki sprejemajo obiske šolarjev. V seznamu se nahajajo nazivi in naslovi. Kmetije imajo predstavitvene strani s kratkimi opisi dejavnosti.

Razvojna agencija Kozjansko^M je objavila seznam 15 (septembra 2014) ekoloških kmetij z nazivi in povezavami na predstavitvene podstrani s kontaktnimi podatki in opisi kmetij.

^G Sinergise d.o.o.: Spletišče Geopedia: <http://www.geopedia.si/>

^H Spletišče zavoda Demeter: <http://www.demeter.si/>

^I Zveza združenj ekoloških kmetov Slovenije (ZZEKS) - Zveza BIODAR: Uporabniki znamke BIODAR: <http://www.zveza-ekokmet.si/uporabniki-znamke-Biodar>

^J Zveza združenj ekoloških kmetov (ZZEKS): Seznam ekokmetij: http://biouzitek.si/BIO_ponudba/Ekoloske_kmetije/%20kraj/

^K Animata d.o.o.: Seznam ekoloških kmetij in trgovin: <http://www.ekomagazin.si/sobi2/>

^L IRT - Inštitut za trajnostni razvoj: Seznam ekoloških kmetij, ki sprejemajo obiske šolarjev: www.itr.si/podrocja_dela/za_sole/kuhinja_sonca/kuhinja_sonca_mapa/postne_stevilke

^M Razvojna agencija Kozjansko: Ekološke kmetije: www.podezelje.com/podezelje/o_podezelje_com/

2.1.3 Prednosti aplikacije

V aplikaciji smo skušali združiti prednosti vseh rešitev, najdenih na spletu, odpraviti nekatere pomanjkljivosti in dodati izpopolnjene možnosti iskanja.

Pregledane obstoječe rešitve, ki ponujajo sezname ekoloških kmetij bodisi preko iskalnika bodisi preko statičnega seznama, ne nudijo naslednjih možnosti, ki so vključene v našo rešitev:

- kraj in čas prodaje neposredno na strani seznama kmetij
- možnost izpisa vseh ponudnikov z vsemi kontaktnimi podatki na eni strani
- alfanumerično razvrščanje po stolpcih, v kolikor ti obstajajo
- kontaktni podatki in povezave na lokacije na zemljevidu neposredno na strani seznama kmetij
- dodatne opcije filtriranja, ki vključujejo:
 - omejitev na regijo in poštni okraj (nekateri rešitve omogočajo eno opcijo, ne pa obeh)
 - omejitev na več blagovnih znamk (nekateri rešitve imajo filter za znamko Biodar)

Pregledane obstoječe rešitve, ki pridelovalcem nudijo možnosti urejanja predstavitvene strani, so bodisi plačniške bodisi ne kažejo znakov interaktivnosti - ni razvidno, ali lahko ponudnik sam ureja svojo predstavitveno stran, ažurira kontaktne podatke.

Druge prednosti naše aplikacije:

- Gre za enega najcenejših oblik trženja kmetijskega gospodarstva. Stroški bodo zgolj na strani organizatorja – zakup spletne domene in najem strežnika za gostovanje.
- Ker gre za spletno predstavitev, je dostopna 24 ur/dan vse dni v letu.
- Verificiranim ekološkim pridelovalcem nudi interaktivnost - lahko se prijavijo v sistem in spremenijo svoje kontaktne podatke, kraj in čas prodaje, določijo ali naj bodo podatki vidni javnosti, določijo ali kmetija sprejema obiske šolarjev ter spišejo opis kmetije, ki je obiskovalcu na voljo na predstavitveni strani kmetije. V opis se lahko doda poljubno vsebino HTML, vključno s povezavami na slike.
- Omogoča več različnih filtrov, ki omejijo rezultate iskanj.
- V rezultatih iskanj dobimo pomembne informacije takoj, brez nepotrebnih dodatnih klikov. To pomeni, da lahko že na strani seznama kmetij izvemo naziv kmetije, kje se kmetija nahaja, kdo je kontaktna oseba, kontaktne podatke, čas in kraj prodaje, katera živila ponujajo.

Slabost aplikacije: Viri podatkov o ponudbi, za katero je kmetovalec pridobil certifikat, se hranijo pri Ministrstvu za kmetijstvo in okolje, ki je podatke pripravilo v obliki, neprimerni za avtomatično posodabljanje z naše strani. Podatki se posodablja 1 x na leto. Upamo na razrešitev tega problema. Optimistična napoved je, da bo aplikacija avtomatiziran mehanizem, ki bo posodabljal podatke o certificiranih ekoloških pridelovalcih brez administratorjevega posega. Slednji bi bil o avtomatično opravljenem delu obveščen preko e-sporočila.^N

^NAvtomatično posodabljanje podatkov v bazi je možno doseči npr. z uporabo t.i. mehanizma Cron Jobs, ki po urniku izvaja naše skripte PHP (<http://docs.cpanel.net/twiki/bin/view/AllDocumentation/CpanelDocs/CronJobs>).

2.2 Učinkovitost aplikacije

Obravnavana aplikacija je učinkovita, če upraviči svoj smisel, oziroma če:

- je za ljudi uporabna oziroma ni le koristna, pač pa ljudje to korist tudi prepoznajo
- je dostopna čim večji množici na več platformah
- je prepoznavna
- znana večjemu krogu ljudi in/ali optimizirana za spletne iskalnike

2.2.1 Uporabnost - koristi aplikacije

- Koristi lahko tistim, ki želijo najti kmetije, ki sprejemajo obiske šolarjev.
- S strani potrošnikov lahko korist v aplikaciji vidijo vsaj tisti, ki si lahko privoščijo (dražja) ekološka živila in si jih ob enem želijo. Poudarek je tudi na intuitivni uporabi aplikacije.
- S strani pridelovalcev lahko koristi tistim, ki si želijo večje prepoznavnosti in tega, da bi stranke lažje prišle do njih. Aplikacija nudi možnost brezplačnega oglašanja na iskalniku in brezplačne predstavitvene strani. Po našem mnenju so ekološke kmetije za trenutno gospodarsko in okoljsko situacijo preveč pomembne, da bi interes objave podatkov s strani pridelovalcev finančno pogojevali.

Ali se lahko krog ljudi, ki bi videli korist v tej aplikaciji, poveča? Zakaj večina ljudi ne kupuje ekološke hrane? Nekateri ne poznajo razlik, nekateri so mnjenja, da jih (skoraj) ni, nekateri nimajo zaupanja v verodostojnost ponudnika. Nekaterim je ta hrana predraga. Da bi se krog ljudi povečal, je torej potrebno:

- Osveščati ljudi o razlikah med ekološkim in konvencionalnim načinom pridelave, prednostih ekološkega načina pridelave. Bistvene razlike in prednosti so predstavljene v prilogi (dodatek 1 in 2). Aplikacija nudi osnovne info o tem.
- Omogočiti, da bo ekološka hrana cenejša. Sistem, ki pripomore k temu, da lahko stranka pride pridelovalcu hrane na dom, omogoča nižjo ceno, saj ni vmesnih členov, ki hrano podražijo (marže preprodajalcev, prispevki za stojnice na tržnici itd.). V tem pogledu je aplikacija koristna.
- Imeti moramo platformo, ki kar se da omeji zlorabe pri prodaji ekoloških pridelkov. Slednje namreč lahko mečejo slabo luč na ekološki način pridelovanja živil, kar lahko zmanjša krog ljudi, katerim bi aplikacija lahko koristila. Zaenkrat nudi aplikacija potrošniku možnost, da se prepriča o veljavnosti ekološkega certifikata. Kot zanimivost tu omenimo še potencialno rešitev – napravo myLapka^o, katero priklopimo na pametni telefon in nam meri, zbira in analizira skrite stvari iz našega okolja. Napovedi so, da bomo lahko napravo uporabili za analizo prisotnosti pesticidov na živilih.

^o Naprava mylapka je predstavljena na spletni strani <http://mylapka.com/>.

2.2.2 Dostopnost aplikacije

Aplikacija je dostopna preko spleta. Preverjeno pravilno deluje na osebnih računalnikih in dlančnikih, ki omogočajo resolucijo nad 700 pikslov širine, preko uporabe spletnih brskalnikov IE, Firefox Mozilla, Google Chrome. S pomočjo Chromove storitve za emulacijo strani na telefonih je bilo preverjeno, da deluje tudi na številnih platformah pametnih telefonov. V bodoče bo razvita različica za pametne telefone, ki bo prilagojena za manjše resolucije zaslona.

2.2.3 Prepoznavnost in všečnost aplikacije

Aplikacija je prepoznavna na račun svoje unikatne celostne grafične podobe (CGP).

»Osnovni elementi celostne grafične podobe so simbol, logotip, črkopis, tipografija, barve, imena in pozicijska gesla.« [6]

»Celostna grafična podoba, vključno z imenom, mora izhajati iz našega poslanstva, vizije ter mora odsevati naš slog, prepričanja, vrednote, ideale in filozofijo. Pri iskanju imena, simbola, logotipa in slogana moramo veliko pozornosti posvečati prav simboliki. Blagovna znamka ni zgolj predmet označevanja, predvsem mora vzbuditi pozornost in zanimanje, željo ter navdušiti za akcijo. Privlačna mora biti za oko in uho, izražati mora osnovno idejo in imeti trajnejšo vrednost, saj le tako lahko postane zvezda stalnica v poslovnem svetu.« [7]

CGP aplikacije za iskanje podatkov o ekoloških kmetijah je usklajena z njeno filozofijo... zelena barva ozadij blokov na spetni strani spominja na zelenje, travnike, ki so značilni za kmetijska gospodarstva. Del ozadja glavnega bloka na prvi strani je zeleno listje. Listi se pojavljajo tudi na gumbih. Prav tako so zeleni gumbi, ikone in tekstovni elementi, ki imajo večji kontrast. Dizajn celotnega spletišča je unikat. Ozadje in ikone so predizajnirane (spremenjene velikosti, barve, teksture, robovi,...) brezplačne grafike.^P

Aplikacija deluje na spletni strani <http://ekološke-kmetije.si>. Ime domene se ujema z imenom spletne strani aplikacije: Ekološke kmetije po Sloveniji.

Zelena barva se velikokrat nanaša na ekologijo. Nekaj primerov iz prakse: »zeleno gospodarstvo«, »zeleni razvojni preboj«, »Zeleni Evrope«. Za temi frazami se skrivajo ekološko usmerjena gibanja.

Pri izdelavi predstavitvenega dela aplikacije so bile upoštevane smernice za dosego večje všečnosti. Barvna shema je enotna, pri slikah so vidne podrobnosti, uporablja se 1 slog pisave (font-family: Helvetica, Arial, 'lucida grande', tahoma, verdana, arial, sans-serif), razmiki med vrsticami in elementi so standardizirani, vsi elementi na strani so poravnani do piksla natančno, tekst je lahko berljiv, kontrastni barvi (zelena in rdeča) sta uporabljeni zgolj na mestih, kjer se želi nekaj posebej izpostaviti.

^P Na spletni strani <http://findicons.com/> je dostopnih mnogo brezplačnih grafik.

2.2.4 Optimizacija aplikacije za spletne iskalnike ter širjenje glasu

Spletni iskalniki (Google, Yahoo, Najdi.si, Bing,...) uporabljajo svojevrstne algoritme, preko katerih določijo, po kakšnem vrstnem redu bodo razvrščene strani na straneh rezultatov iskanj za vpisano iskalno frazo (iskalni niz, ki ga vnese uporabnik). Optimizacija strani za spletne iskalnike - SEO (Search Engine Optimization) je vrsta načinov, ki lahko pomagajo izboljšati pozicijo specifičnega spletišča na straneh rezultatov spletnih iskalnikov. Da bi bil proces SEO uspešen, je potrebna natančna analiza iskalnih fraz, ki so povezane s spletiščem, katerega želimo optimizirati.

Raziskati je potrebno, koliko ljudi išče po teh frazah na spletnih iskalnikih in/ali strateško načrtovati promoviranje določenih ključnih besed preko različnih medijev in nato optimizirati spletišče za te ključne besede (primer... če se v medijih kar naprej oglašava beseda avatar, lahko pričakujemo, da bo po tej besedi začelo iskati veliko ljudi... posledično pa je dobro našo stran optimizirati za to ključno besedo - avatar).

Ključne besede so besede na spletni strani ali spletišču, ki so zanj(o) najbolj pomembne. Optimalno je, da na spletno stran umestimo takšne ključne besede, po katerih išče veliko število ljudi, hkrati pa besede niso preveč konkurenčne (niso optimizirane na drugih straneh). Poleg tega vključuje SEO tudi vsebinske in tehnične popravke na spletni strani, gradnjo zunanjih povezav,...

Posledica tega, da se spletišče za izbrane ključne besede pojavi na prvih straneh rezultatov spletnih iskalnikov, je večja obiskanost spletišča. Skrt [8] razlaga, da se po nekaterih raziskavah na prvi strani rezultatov iskanj zgodi cca 79% vseh klikov, na drugi strani 10% in na tretji 11%. Če spletna stran za pomembne ključne besede ni dovolj dobro optimizirana, oziroma če je spletni iskalnik ne prepozna kot dovolj pomembno, bodo do nje preko spletnega iskalnika prišli le redki obiskovalci. Skrt [8] navaja tudi, da se svetovno gledano večinoma uporablja iskalnik Google (71%), nato Yahoo (18%). Pri [9] navajajo, da sta v Sloveniji najbolj priljubljena spletna iskalnika Google in Najdi.si.

Za SEO je pomembno, da poskrbimo, da bo pomembna vsebina strani dostopna za spletnega robota (pajka, angl. Web crawler ali web spider ali automatic indexer) spletnega iskalnika. Na vsake toliko časa (odvisno od iskalnika in tega, kolikokrat prihaja do sprememb na strani) spletni iskalniki pošljejo na spletišče (če se nahaja v njihovi bazi, ali če pridejo do njega preko neke povezave iz drugega spletišča) svojega robota, na račun katerega posodobijo vsebino indeksov vsebin drugih spletišč.

Pajek začne s seznamom URL povezav. Obišče strani, na katere kažejo povezave in na teh straneh poišče hiperpovezave (hyperlink). Slednje doda na seznam URL povezav za kasnejšo obdelavo. Na tak način rekurzivno obišče vse povezave spletišč (nekateri iskalniki ignorirajo povezave z atributom no-follow in določene mime tipe datotek, do katerih vodijo povezave). Pajek lahko skopira celotno vsebino obiskane strani za kasnejše procesiranje s strani algoritma iskalnika, ki indeksira vsebino z namenom, da bi lahko iskanje po njih potekalo hitreje.

Podrobnejše delovanje pajka si lahko bralec ogleda v [10]. Grafična podoba strani spletnih iskalnikov načeloma ne zanima, saj spletno stran vidijo povsem drugače, kot jo vidimo ljudje.^Q

Za indeksiranje iskalniki uporabljajo podatkovno strukturo drevo ali pa razpršeno tabelo. Indeksiranje je organiziranje več milijonov besed v podatkovno strukturo s takšno hierarhijo, ki omogoča, da je hitrost iskanja besed največ logaritmčna. Iskalniki med indeksiranjem dodajajo besedam, zastopanim na strani, posebno težo glede na njihovo pogostost in glede na to ali jih prepozna kot ključne besede.

V splošnem velja, da preko spletnih iskalnikov obišče spletne strani večina obiskovalcev, zato je za boljšo obiskanost strani pomembno upoštevati nekaj smernic, kako optimizirati spletišče in kako izpostaviti ključne besede na spletišču. O SEO sicer obstaja veliko knjig, ampak algoritmi spletnih iskalnikov se ves čas spreminjajo. Glede na to, da sta za slovensko območje pomembna predvsem iskalnika Google in Najdi.si, so bila pri optimizaciji upoštevana njihova navodila [11] in [12]. SEO poudarek aplikacije je na:

- Velikosti spletišča... velikost izraža število podstrani oziroma notranjih povezav ter količino besed na posamezni strani. V kolikor na neko stran kaže več notranjih povezav, jo spletni iskalnik prepozna za bolj pomembno in ji dodeli višji rang. Velikost spletišča narašča na račun podstrani - predstavitev strani pridelovalcev ekoloških živil. Vsak pridelovalec ima eno predstavitevno stran.
- Unikatnosti vsebine... nekateri spletni iskalniki kaznujejo strani, ki so duplikati drugih strani. Na predstavitevni strani kmetij opisi zato niso bili kopirani z referenčnih strani.
- Strukturi strani... za diplomsko nalogo je pomembno, da je na začetku naslov, potem povzetek, uvod,... Enako velja za spletno stran. Pomembna je hierarhija, označevanje glavnih naslovov, podnaslovov, pričakuje se, da so bistva izpostavljena na začetku. Ključne besede, ki so na strani v samem začetku, so pomembnejše.
- Specializaciji... ena stran naj bo optimizirana na 2-3 ključne besede oziroma nize. Te besede v vsebini strani okrepimo ali podčrtamo.
- Hitrosti nalaganja vsebine strani... dobro je, da se slike komprimira, uporabi CSS sprites. Na hitrost nalaganja vplivajo tudi tehnične značilnosti strežnika, itd.
- Konsistentnosti med vsebino strani in parametri v glavi strani (meta ključne besede, naslov, opis)... nekateri spletni iskalniki kaznujejo poskuse zlorabe algoritmov iskalnikov z vključevanjem neustreznih ključnih besed v glavo strani. Primer: nekdo v vsebini strani promovira viagro, v meta oznake pa zapiše, da se stran nanaša na ekološka živila.
- Čim večjim številom vhodnih povezav (povezav s strani drugih spletišč) in čim večjim rangom teh strani... s tem narašča rang strani. Rang bistveno vpliva na pozicijo spletne strani na straneh rezultatov iskanj. V bodoče bo naš poudarek na pridobivanju povezav od zunaj, širjenju glasu preko letakov na ekoloških sejnih, ekoloških tržnicah in ustnem izročilu.

^Q Kako pajek vidi spletno stran, si lahko ogledamo na strani <http://lynx.isc.org>.

- Besedah v povezavah, ki vodijo do neke strani, ki jo želimo optimizirati... te besede morajo biti zastopane na strani, katero optimiziramo. Dobro je, da so besede v povezavah ključne besede na strani, ki jo optimiziramo.
- Starosti domene in časom zakupa domene... domena, ki je starejša, je bolj kredibilna. Če je domena zakupljena za daljši rok (več let), lahko v določenih primerih pri spletnih iskalnikih izpade, da je vsebina bolj resna in da recimo za tem stoji organizacija oziroma nekdo, ki nudi pomembnejše informacije. Zaradi finančnih razlogov je bila domena ekološke-kmetije.si registrirana za 2 leti.
- Primernem izboru vsebine meta oznak v glavi strani. Poskrbeli smo, da imajo vse javne strani spletišča primerne unikatne meta oznake.

Na spletu lahko poleg teh priporočil najdemo sezname več kot stotih^R različnih parametrov, ki (lahko) vplivajo na končno pozicijo neke strani na straneh rezultatov spletnih iskalnikov. Včasih pri ponudnikih storitev SEO naletimo tudi na zavajanja. Google na primer je že večkrat objavil, da meta ključnih besed v glavi strani pri določanju pozicije strani ne obravnava več, pa bomo pri nekaterih ponudnikih spletnega optimiziranja še vedno naleteli na trditve, da jih obravnava.

Spletna stran iskalnika ekoloških kmetij je unikatna tako po grafiki, programski kodi, kot tekstovni vsebini in je optimizirana za nekatere ključne besede, ki so najbolj povezljive s tem, kar iskalnik ekoloških kmetij nudi. Poskrbljeno je, da je urejena struktura strani, hitrost nalaganja, konsistentnost vsebine in ključnih besed.

Spletni pajek preko obiska indeksne strani iskalnika ekoloških kmetij ne bo prišel do podstrani kmetij, ki imajo povratno povezavo na indeksno stran, ker se povezave do njih naložijo dinamično – preko poizvedbe Ajax. To težavo smo rešili s pomočjo datoteke `sitemap.xml`, ki jo spletni pajek obišče ob vsakem prihodu na spletišče. Datoteka predstavlja pajku seznam strani na spletišču, ki jih je potrebno obiskati.

Zgled preproste datoteke `sitemap.xml` za glavno stran in dve podstrani:

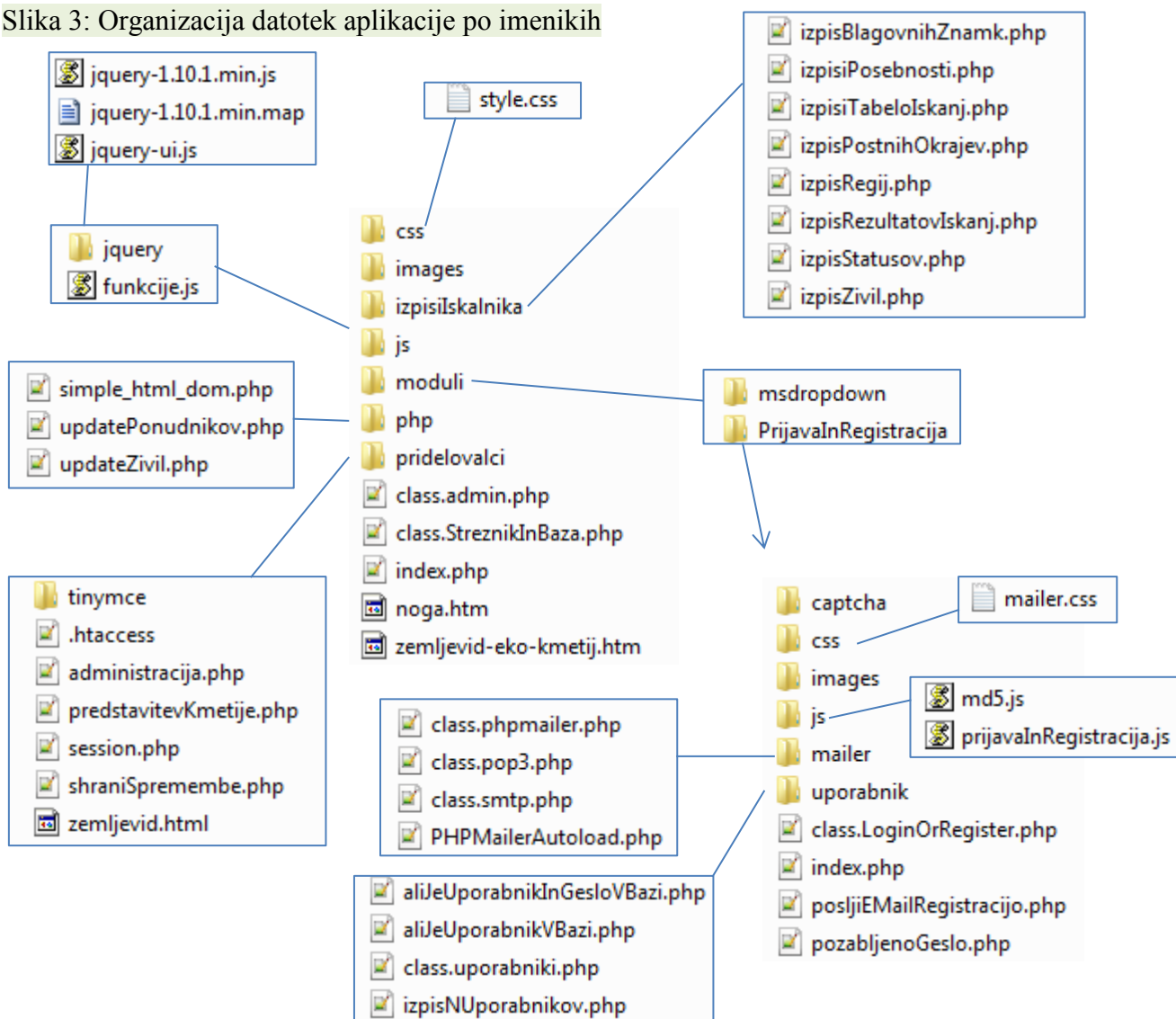
```
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sitemaps.org/schemas/sitemap/0.9
http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd">
  <url><loc>http://www.ekološke-kmetije.si/</loc></url>
  <url><loc>http://www.ekološke-kmetije.si/pridelovalci/kmetija-
ronkaldo</loc></url>
  <url>
    <loc>http://www.ekološke-kmetije.si/pridelovalci/ambrozic-lidija</loc>
  </url>
</urlset>
```

^R : Na strani <http://www.webconfs.com/15-minute-seo.php> je prispevek z naslovom »List of Best and Worst practices for designing a high traffic website.«. V njem je navedenih 77 parametrov, ki vplivajo na pozicijo spletne strani. Nanašajo se na ključne besede, povezave, meta oznake, vsebino, vizualne vsebine, domeno, zasnovo URL.

3 Razvoj aplikacije

Za lažje razumevanje delovanja aplikacije si najprej na sliki 3 oglejmo katere imenike in datoteke vsebuje.

Slika 3: Organizacija datotek aplikacije po imenikih



Opis imenikov in datotek:

- `Index.php` je glavna datoteka iskalnika, opisana v podpoglavju 3.3.3.
- `class.admin.php` in `class.StreznikInBaza.php` sta dva razreda, ki nam omogočata boljšo organizacijo. Prvi je opisan v podpoglavju 3.3.4.2, drugi pa v podpoglavju 3.3.2.
- `noga.htm` je datoteka za izpis noge strani in se uporablja pri vseh straneh.
- `Zemljevid-eko-kmetij.htm` vsebuje kodo za izpis zemljevida vseh kmetij v elementu HTML `<iframe>`. Zgled prikaza zemljevida je podan v podpoglavju 3.4.4.
- Imenik `css` hrani stilsko datoteko `style.css`. Za razumevanje aplikacije ni potreben opis.
- Imenik `images` vsebuje uporabljene slike. Opis ni potreben.

- Imenik izpisIskalnika vsebuje datoteke, ki jih uporablja iskalnik. Datoteko `izpisiRezultatovIskanj.php` uporablja za izpis rezultatov iskanj, zato je podrobno opisana v podpoglavju 3.4.2.1.2. Ostale datoteke se uporabljajo za prikaz vsebine filtrov. Kliče jih datoteka `index.php`, ki je podrobno opisana v podpoglavju 3.3.3.
- Imenik js vsebuje datoteke s kodo JavaScript. V podimeniku jquery so tuje knjižnice, omenjene v podpoglavju 3.4.7. Datoteka `funkcije.js` vsebuje večinoma lastno razvite funkcije, ki se uporabljajo za manipulacijo prikaza spletne strani (dogodki ob premikanju drsnika in spreminjanju velikosti okna, skrivanje objektov, klicanje poizvedb Ajax, preprečevanje zapuščanja strani po pomoti, lep prikaz opisnega besedila na filtrih, animacijo osla, prikazovanje in skrivanje oken na zahtevo (okno za navodila uporabe, okno za prikaz zemljevida, okna za opis logotipov na dnu zaslona), funkcije za izpis rezultatov iskanj, funkcije za izpis arhiva iskanj, funkcije za razvrščanje stolpcev v rezultatih iskanj, funkcije za shranjevanje vsebine na administrativnih straneh uporabnikov). V podpoglavju 3.4.2.1.1 je opisano delovanje poizvedbe Ajax za izpis rezultatov iskanj.
- Imenik moduli vsebuje 2 modula:
 - Msdropdown je tuj modul, opisan v podpoglavju 3.4.7.
 - PrijavaInRegistracija je modul, ki je v celoti razvit z naše strani in se uporablja za izpis forme za prijavo in registracijo. Datoteka `prijavaInRegistracija.js` vsebuje funkcije, ki skrbijo za obravnavo dogodkov ob uporabi form. Preko klicev Ajax se kličejo datoteke PHP iz podimenika uporabniki ali pa datoteki iz osnovnega imenika: `posljiEmailRegistracijo.php` in `pozabljenoGeslo.php`. Koda modula ne po posebej obravnavana, ker to ni pomembno za temo diplomskega dela. Delno je obravnavan zgolj princip vzpostavljanja seje (session) ob uspešni prijavi in zapiranju seje ob odjavi. Več v podpoglavju 3.4.6.2. Pomembnejše od kode je razumevanje logike registriranja uporabnika, ki mora hkrati že obstajati v sistemu. To obravnavamo v podpoglavju 3.4.5. Omenimo le še to, da modul uporablja knjižnico (re)captcha za onemogočanje spamma ob registraciji, knjižnice mailer za prilagojeno formatiranje e-sporočil in knjižnico md5 za kriptiranje gesel. Več o tem v podpoglavju 3.4.7.
- Imenik php vsebuje skripto `updateZivil.php`, sprogramirano za posodobitev živil v bazi (glej podpoglavje 3.3.4.15) in skripto `updatePonudnikov.php`, ki je opisana med dodatki. Slednja se zna povezati na več tujih spletnih podstrani, z njih prebrati uporabne podatke o pridelovalcih ekoloških živil in jih shraniti v našo podatkovno bazo.
- Imenik pridelovalci vsebuje skripte PHP za izpis predstavitev strani kmetij in izpis administrativne strani registriranega uporabnika. Delovanje je opisano v podpoglavju 3.4.6. Podimenik tinymce je modul, ki uporabniku omogoča, da spiše opis kmetije preko naprednega uporabniškega vmesnika, ki omogoča vnos besedila HTML. Opisan je v podpoglavju 3.4.7.

3.1 Tehnologije, razvojna in programska orodja

Aplikacija je napisana v jezikih HTML, JavaScript, PHP, SQL, CSS. Koda je bila spisana v programu Notepad ++^S, tekom programiranja se je preizkušala predvsem v spletnem brskalniku Google Chrome^T. V veliko pomoč pri programiranju je bila njegova funkcionalnost »Preglej element«, ki omogoča vpogled v kodo HTML za poljubni element spletne strani, nadzor kode Java Script v času interpretiranja njenih ukazov, prilagajanje stila strani preko kode CSS, itd.

Osprednji del (Frontend) je stran HTML s formami, opremljena z funkcijami JavaScript in oblikovnimi določili CSS. Strežniški del aplikacije (Backend) je sprogramiran v PHP in PHP-5 z objektno usmerjenim programiranjem. Podatki se shranjujejo v podatkovni bazi MySQL.

HTML (Hyper Text Markup Language) je označevalni jezik za izdelavo spletnih strani. HTML ni programski jezik. Izvaja se na spletnem brskalniku.

JavaScript je objektni skriptni programski jezik, ki sodeluje s kodo HTML in jo nadgradi z dinamičnim izvajanjem, na primer premikom specifičnega elementa spletne strani. Koda JavaScript je lahko vgrajena neposredno med posamezne elemente v telesu dokumenta HTML, ali pa je vključena kot datoteka. Začetek kode JavaScript neposredno med kodo je podan z nizom `<script type='text/javascript'>`, konec pa z `</script>`. JavaScript nudi interaktivnost med uporabnikom in spletno stranjo na nivoju lokalnega računalnika (vse se izvaja na strani odjemalca).

PHP je strežniški skriptni jezik, ki se lahko uporablja za izdelavo spletnih strani, ki so v interakciji s spletnim strežnikom med njeno uporabo na strani odjemalca. V programu pišemo kodo PHP v kombinaciji s HTML. Začetek kode PHP je določen z nizom `<?>` ali `<?php>`, konec pa z `>`. Tak niz lahko vrinemo kamorkoli med kodo HTML. Ukazi, ki so med tema dvema nizoma, se bodo izvedli na spletnem strežniku, ne na strani odjemalca. Za to skrbi interpreter PHP, ki je običajno implementiran kot modul v jedru spletnega strežnika, ali pa kot izvajalni vmesnik CGI^U.

MySQL je odprto kodni sistem za upravljanje relacijskih podatkovnih zbirk. Kratica SQL pomeni strukturiran poizvedbeni jezik (Structured Query Language). SQL je programski jezik, v katerem pišemo kodo za manipulacijo podatkov, shranjenih v relacijski podatkovni zbirki (na primer kodo za dostop do podatkov tabel ali spreminjanje vsebine specifičnih vrstic, stolpcev, ipd.). Poizvedbe se izvedejo na strežniku MySQL.

Aplikacija se je razvijala na osebnem računalniku, ki je bil opremljen z odprto kodnim programskim paketom AMPPS^V, ki vključuje:

- Spletni strežnik Apache
- Podatkovno bazo MySQL in aplikacijo phpMyAdmin za administriranje baz, tabel,...
- Programsko opremo za izvajanje spletnih skript (PHP, Perl, Python)

^S Spletna stran napredne beležnice Notepad++: <http://notepad-plus-plus.org/>

^T Spletna stran brskalnika Google Chrome: <http://www.google.com/intl/sl/chrome/>

^U PHP kot CGI ali kot modul: <http://www.tuxradar.com/practicalphp/18/6/0>

^V Spletna stran paketa AMPPS: <http://www.ampps.com/>

3.2 Nastavitev jezika

Spletne strani aplikacije so na voljo v slovenskem jeziku. Uporablja se kodna tabela UTF-8, ki podpira slovenske šumnike. Da ne pride do kakšnih dodatnih znakov, ki bi prelamljali vrstice spletne strani, so vse datoteke shranjene z urejevalnikom Notepad ++, ki podpira shranjevanje datotek v načinu UTF-8 brez BOM znaka.

Da bi spletni brskalniki obiskovalcev strani prepoznali, da spletna stran servira podatke v obliki UTF-8, je v glavi strani nastavljena sledeča oznaka:

```
<meta http-equiv="Content-type" content="text/html; charset=UTF-8" />
```

Spletni brskalnik, ki ima na primer za kodiranje izbran Windows-1250, bo črko ž izpisal kot Ĺĺ. Kodiranje UTF-8 pa jo prikaže v pravilni obliki.

Strežnik odjemalcu (spletnem brskalniku) v spremenljivki »Content-Type header« pošlje informacijo o načinu kodiranja. Privzeto kodiranje na strežniku se lahko razlikuje od tega, ki ga mi potrebujemo, zato lahko nastavimo želeno kodiranje sami. Posledica spodnjega ukaza je, da lahko na strežniku pravilno prikazujemo šumnike.

```
<?php header('Content-Type: text/html; charset=utf-8'); ?>
```

Paziti je treba, da pred tem ukazom ni elementov HTML, sicer pride do napak.

Ali je datoteka shranjena v UTF-8, lahko preverimo na naslednji način:

```
<?php $text = file_get_contents("naša_datoteka.php"); ?>
<?php if (preg_match('/./u', $text) == true) { echo "UTF-8 način"; } ?>
```

Aplikacija uporablja tudi funkcije za razvrščanje podatkov. Zato je potrebno nastaviti še lokalno spremenljivko, da strežnik pri uporabi sort funkcije ve, kako razvrščati znake.

```
<?php setlocale (LC_ALL, 'sl_SI.UTF-8'); //Za vse kategorije ?>
<?php setlocale (LC_TIME, 'sl_SI.UTF-8'); //Za datumske funkcije ?>
```

Te funkcije lahko seveda delujejo le v primeru, če je na strežniku nameščen jezikovni paket za slovenski jezik. V operacijskem sistemu Linux lahko to preverimo tako, da v terminal vpišemo ukaz `sudo locale -a`. V primeru, da v izpisu ne najdemo ustreznega jezika, moramo namestiti programski paket z ukazom `sudo /usr/share/locales/install-language-pack sl_SI`.

Podatkovna baza je v strukturi UTF-8. Pravilo za razvrščanje znakov v tabelah in stolpcih v posameznih tabelah je nastavljeno na `utf8_slovenian_ci`, da lahko razvrščamo rezultate iskanj po abecednem vrstnem redu. Pri poizvedbah SQL povemo mehanizmu PHP, da naj prenaša podatke v načinu UTF:

```
<?php mysql_query("SET NAMES 'utf8' COLLATE 'utf8_slovenian_ci'"); ?>
<?php mysql_query("SET CHARSET SET 'utf8_slovenian_ci';"); ?>
```

3.3 Dostop do podatkov

Obravnavamo dva tipa dostopa do podatkov. Prvi se nanaša na to, kako bo do podatkov spletne strani prišel spletni brskalnik, drugi pa na to, kako bo skripta PHP na strežniku dostopila do podatkov v podatkovni bazi.

3.3.1 Dostop do podatkov spletne strani s strani brskalnika

IP (Internet Protocol) je osnovni komunikacijski protokol za pošiljanje podatkov preko omrežja.

Ko vtipkamo naslov do spletne strani aplikacije, se odjemalec (brskalnik) poveže s spletnim strežnikom in od njega prejme podatke. Proces poteka po korakih [13]:

1. Odjemalec se preko omrežne kartice poveže s strežnikom domenskih imen (strežnik DNS) in mu pošlje ime domene, do katere želi dostopiti. Ime domene je del spletnega naslova. Strežnik DNS posreduje odjemalcu številko IP računalnika (spletnega strežnika), na katerem je shranjena spletna stran.
2. Odjemalec preko vtičnice IP odpre povezavo s tem IP naslovom.
3. Skozi vtičnico zapiše podatkovni tok HTTP, v katerem je navodilo strežniku naj mu pošlje spletno stran v kodi HTML.
4. Spletni strežnik odpre datoteko index.php (ali katero drugo) in po vrsti interpretira ukaze, zapisane v njej. V našem primeru gre za različne funkcije, ki izvedejo poizvedovanje po podatkovni bazi. Rezultat je koda spletne strani, zapisana v jeziku HTML. To kodo pošlje spletni strežnik odjemalcu kot podatkovni tok HTTP.
5. V primeru, da pride do napake (na primer, da je strežnik prezaseden ali da je nekaj narobe z zahtevo ali kaj tretjega), pošlje spletni strežnik odjemalcu statusno kodo napake (na primer 400, 401, 402, 404... , 500, 501,...). Na primer: Če stran, do katere dostopamo, ne obstaja, strežnik vrne kodo napake 404.
6. Podatkovni tok HTTP se na računalniku odjemalca shrani kot datoteka HTML v imeniku, kamor se shranjujejo začasne datoteke. Brskalnik to datoteko odpre in interpretira vsebino strani.

3.3.2 Dostop do podatkov podatkovne baze s strani skripte PHP

Podoben proces, kot zgoraj (prvi korak ni več potreben), se izvede, ko obiskovalec spletne strani uporabi formo za prijavo ali registracijo ali pa formo iskalnika na tej strani. Razlika je v tem, da tokrat spletni strežnik od odjemalca prejme v podatkovnem toku HTTP še določene podatke o filtrih, ki so podani kot parametri akcije spletne forme. Na strani strežnika so podatki skripti PHP, ki te podatke uporabi pri generiranju poizvedb, dostopni preko spremenljivke `$_POST` ali `$_GET` (odvisno od tega, na kateri način so bili posredovani).

Pred dostopom do baze se mora spletni strežnik najprej povezati s podatkovno bazo. Ker uporabljamo 11 datotek, preko katerih se povezujemo z bazo, je bila za nastavitve strežnika MySQL in baze ustvarjena datoteka `class.StreznikInBaza.php` (predstavljena na naslednji strani), v kateri funkcija `vrniKonfiguriranoBazo()` ustvari nov objekt tipa `StreznikInBaza`, ki poenostavi konfiguracijo.

```

<?php //Vsebinska datoteke class.StreznikInBaza.php
function vrniKonfiguriranoBazo() { //Vrne instanco razreda StreznikInBaza.
    $hostG = "localhost";
    $bazaG = "EkoPonudba";
    $upImeG = "admin";
    $gesloG = "";
    $jezikG = "sl-SI";
    return new StreznikInBaza($hostG, $upImeG, $gesloG, $bazaG, $jezikG);
}
class StreznikInBaza { //Razred za nastavitev strežnika in baze.
    private $povezava;
    private $host;
    private $baza;
    private $upIme;
    private $geslo;
    private $jezik;
    //Konstruktor
    public function StreznikInBaza($host, $upIme, $geslo, $baza, $jezik) {
        $this->host = $host;
        $this->upIme = $upIme;
        $this->geslo = $geslo;
        $this->nastaviPovezavo();
        $this->$baza = $baza;
        $this->nastaviBazo($baza);
        $this->nastaviJezik($jezik);
        //Nastavimo najdaljši možni čas za odziv strežnika. 300 sekund.
        ini_set('max_execution_time', 300);
    }
    public function nastaviPovezavo() {
        $this->povezava = mysql_connect($this->vrniHost(), $this->vrniUpIme(),
            $this->vrniGeslo()) or die (mysql_error());
    }
    public function nastaviBazo($baza) {
        mysql_select_db($baza); //izberemo podatkovno bazo, za delo
    }
    public function nastaviJezik($jezik) {
        $this->jezik = $jezik;
        if ($jezik == "sl-SI") {
            setlocale (LC_ALL, 'sl_SI.UTF-8'); //za vse kategorije
            setlocale (LC_TIME, 'sl_SI.UTF-8'); //za datumske funkcije
            mysql_query("SET CHARACTER SET 'utf8_slovenian_ci'");
            mysql_query("SET NAMES 'utf8' COLLATE 'utf8_slovenian_ci'");
            mysql_query("SET COLLATION_CONNECTION='utf8_slovenian_ci'");
        }
    }
    //Funkcije, ki vračajo vrednosti instance objekta StresnikInBaza
    public function vrniPovezavo() { return $this->povezava; }
    public function vrniHost () { return $this->host; }
    public function vrniUpIme () { return $this->upIme; }
    public function vrniGeslo () { return $this->geslo; }
    public function vrniJezik () { return $this->jezik; }
}??

```


V datoteki, v kateri želimo uporabljati ta objekt, najprej vključimo datoteko `class.StreznikInBaza.php`. Nato ustvarimo spremenljivko, kateri dodelimo vrednost s klicem funkcije `vrniKonfiguriranoBazo()`:

```
<?php include("../class.StreznikInBaza.php"); ?>
<?php $setupDB1 = vrniKonfiguriranoBazo(); ?>
```

Na koncu datoteke, ki uporablja razred, je dobro, da prekinemo povezavo s podatkovno bazo:

```
<?php mysql_close($setupDB1->vrniPovezavo());
```

Datoteka `class.StreznikInBaza.php` pride zelo prav v primeru, da prestavimo aplikacijo na nek drug spletni strežnik. Na račun te datoteke podatke za dostop do baze nastavimo samo na enem mestu.

3.3.3 Vsebina datoteke `index.php`

Ko se odjemalec (spletni brskalnik) poveže s spletnim strežnikom na naslovu iskalnika ekoloških kmetij, začne strežnik interpretirati ukaze znotraj datoteke `index.php`. V njej so sklici do vseh datotek, ki jih iskalnik uporablja. Za tem, ko se koda PHP na strežniku izvede, pošlje spletni strežnik rezultat odjemalcu. V našem primeru se pošlje indeksna spletna stran, pretvorjena v obliko HTML, vse datoteke CSS in JavaScript, ki jih ta stran uporablja in vse slike.

Osnovna spletna stran HTML izgleda takole: [14]

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<h1>Naslov</h1>
<p>Odstavek</p>
</body>
</html>
```

V glavi `<head>` dokumenta HTML se nahajajo [15]:

- Element `<title>`, v katerem je podan naslov strani
- Element `<base>`, ki specificira osnovni ciljni URL za vse relativne URLje spletišča. Če nastavimo cilj (njegov parameter `target`) na `blank`, se bodo vse notranje povezave odprle v novem zavihku.
- Element `<link>` definira relacijo med dokumentom in eksternimi viri, kot je datoteka CSS, katere vsebina so navodila brskalniku o tem, v kakšni obliki in na katerih mestih naj postavi specifične elemente na strani,...
- Element `<style>` uporabljamo za neposredno definiranje stilskih informacij o spletni strani.
- Element `<meta ... >` ... podatki o podatkih na spletni strani. Navodila spletnim iskalnikom, brskalnikom in drugim spletnim servisom. Oznake meta specificirajo opis strani, ključne besede, uporabljene na strani, avtorja dokumenta, čas zadnje spremembe,...

Spodaj si lahko ogledamo, kako izgleda začetni del datoteke `index.php`.

```
<?php
header('Content-Type: text/html; charset=utf-8'); //Glej podpoglavje 3.2

//Vključimo datoteko za nastavitev strežnika, baze, jezika
include("../class.StreznikInBaza.php");

//Ustvarimo instanco razreda StreznikInBaza.
$setupDB1 = vrniKonfiguriranoBazo();//da bomo lahko izvajali poizv. SQL.
?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <!--Oznake meta -->
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <meta name="description" content="Ekološke kmetije po Sloveniji -
        iskalnik ekoloških kmetij. Zemljevid in spisek kmetij z veljavnim
        ekološkim certifikatom." />
    <meta name="Keywords"
        content="kmetija, kmetije, ekološki, biološki, certifikat,
        ekološke kmetije, ekološka kmetija, zemljevid ekoloških kmetij,
        iskalnik ekoloških kmetij, spisek ekoloških kmetij" />
    <title>Ekološke kmetije - iskalnik pridelovalcev ekoloških živil</title>

    <!--Vse stilske oznake iskalnika so zbrane v eni datoteki CSS -->
    <link href="./css/style.css" rel="stylesheet" type="text/css" />

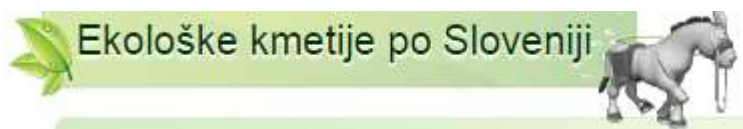
    <!--Knjižnica jQuery, opisana v podpoglavju 3.4.7 -->
    <script src="./js/jquery/jquery-1.10.1.min.js"></script>
    <script src="./js/jquery/jquery-ui.js"></script>

    <!--Modul msdropdown, opisan v podpoglavju 3.4.7 -->
    <link rel="stylesheet" type="text/css" href="./css/msdropdown/dd.css" />
    <script src="./js/msdropdown/jquery.dd.min.js"></script>
    <!--Konec msdropdown -->

    <!--Modul za prijavo, registracijo, pozabljena gesla, e-pošto -->
    <!--Modul je opisan v podpoglavju 3.4.5. -->
    <script src="./PrijavaInRegistracija/js/md5.js"
        type="text/javascript">
    </script>
    <link rel="stylesheet" type="text/css"
        href="./PrijavaInRegistracija/css/mailer.css" />
    <script type="text/javascript"> var baseDir = "."; </script>
    <!--Konec modula -->

    <!--Knjižnica Recaptcha za antispam zaščito pri registraciji -->
    <!--Uporablja jo modul PrijavaInRegistracija, oziroma modul reCaptcha-->
    <script type="text/javascript"
        src="http://www.google.com/recaptcha/api/js/recaptcha_ajax.js">
    </script>
</head>
```

Slika 4- naslov dokumenta - <H1>



podanih v datoteki `style.css`. Za spletni iskanik (Google ipd.) je pomembno, da besedila s ključnimi besedami, ki so za stran bolj pomembne (ker obiskovalci po njih pogosteje iščejo), postavimo čim bolj na vrh dokumenta.

Na prvem mestu se nahaja naslov `<h1>`, (glej sliko 4) katerega na zaslon pripelje osel. Animacija je kombinacija gif slike in metode jQuery za translacijo bloka `<div>`, v katerega je postavljen naslov in slika.

Temu sledi naslov `<h2>Iskalnik ekoloških kmetij</h2>` in potem opis ekološkega kmetovanja. V opisu so pomembne ključne besede označene s krepko pisavo.

Slika 5: Blok z iskalnimi polji in blok z opisom ekološkega kmetovanja

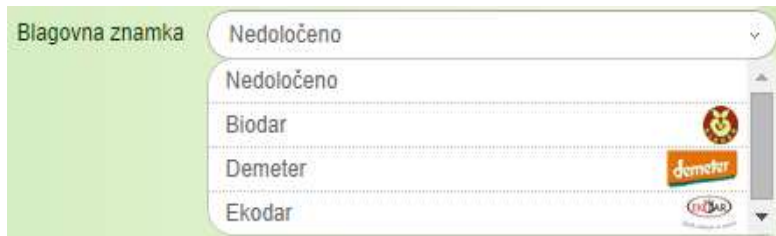
Slika 5 prikazuje, kako izleda blok z vsebino forme in opisom ekološkega kmetovanja. Spletni pajek (glej podpoglavje 2.2.4), ki kopira vsebino, katero kasneje iskalnik indeksira, ne ve, kako je vsebina prikazana na zaslonu. Kopira samo besedilo. Čeravno smo v Sloveniji navajeni brati z leve proti desni in so vnosna polja našega iskalnika po takšni logiki na prvem mestu, pa v dokumentu niso postavljena na prvo mesto, saj v njih ni tako dobrih ključnih besed. Takšen maneuver nam omogoča CSS - poljuben blok lahko postavimo na poljubno mesto na zaslonu ne glede na lokacijo v dokumentu HTML. Blok z iskalnimi polji je v resnici za blokom z opisom ekološkega kmetovanja.

Iskalna polja blagovna znamka, regija, poštni okraj, status kmetije in ponudba kmetije so sezname (elementi `<select>`) z več opcijami. Opcije `<option>` znotraj elementa `<select>` vsebujejo dodatne atribute, ki jih potrebuje knjižnica msdropdown (glej podpoglavje 3.4.7). Zaradi boljše organizacije je koda za izpis vsakega filtra zapisana v svoji datoteki tipa `.php` ali `.htm`. Datoteke za izpis podatkov filtrov in izpis podatkov iskanj so shranjene v

podimeniku `./izpisiIskalnika`. Ko datoteke vključimo v dokument `index.php` s funkcijo `include()`, se izpiše koda posameznih polj in njihovih opcij. Zgled:

```
<?php include('izpisiIskalnika/izpisBlagovnihZnamk.php'); ?>
```

Slika 6: Primer iskalnega polja za omejitev izpisa glede na blagovno znamko



Zgornji ukaz kliče datoteko `izpisBlagovnihZnamk.php`.

Spletni strežnik začne interpretirati njeno vsebino, ki vrne podatke, katere se prikažejo v filtru »Blagovna znamka« (glej sliko 6):

Vsebina datoteke `izpisBlagovnihZnamk.php`:

```
<?php
$query= "SELECT ime_logotipa, id_logotipa, ime_slike FROM Logotip where
bool_blagovna_znamka=1 order by id_logotipa";
if ($slike = mysql_query($query)) {
    echo('<select name="blagovnaZnamka"
        class="txt-jumpmenu"
        id="blagovnaZnamka"
        onchange"$ ("#blagovnaZnamka").blur();
    ">');
    echo('<option value="0"
        data-image="./images/logotipi/blank.png"
        data-imagecss="small-image"
        data-description="Nedoločeno" >. </option>');
    while($row = mysql_fetch_assoc($slike)) {
        echo('<option value="'. $row["id_logotipa"].'"
            data-image="./images/logotipi/'. $row["ime_slike"].'"
            data-description="'. $row["ime_logotipa"].'"
            data-imagecss="small-image"
            data-title="">.</option>'
        );
    }
    echo("</select>");}else{echo (mysql_error ());}
?>
```

Slika 7: Vrednosti , ki jih izpiše poizvedba SQL za izpis blagovnih znamk

ime_logotipa	id_logotipa	ime_slike
Biodar	1	biodar.png
Demeter	5	demeter.png
Ekodar	9	ekodar.png

Opis zgornje kode: Funkcija `mysql_query` izvede poizvedbo, shranjeno v spremenljivki `$query`, katero posredujemo funkciji v argumentu. V našem primeru vrne podatke, ki jih prikazuje slika 7.

Te podatke potem izpišemo v formatirani obliki s pomočjo funkcije `echo`. V našem primeru izpišemo kodo filtra blagovna znamka na tisto mesto v dokumentu, kjer smo s funkcijo `include` klicali dokument `izpisBlagovnihZnamk.php`. Interpreter php na tak način postopno sestavlja nov dokument HTML, ki se v končni fazi pošlje odjemalcu.

Na podoben način naložimo tudi ostale filtre.

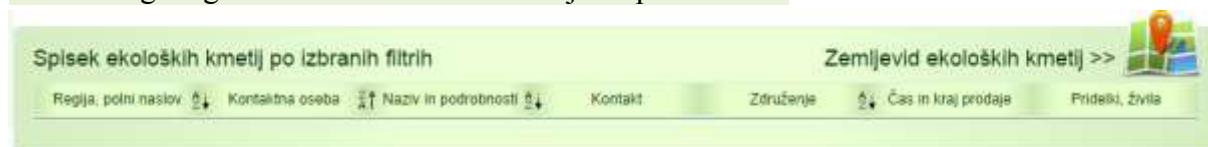
Vsebina zemljevida se ne naloži ob dostopu do strani iskalnika, ampak kasneje na zahtevo – s klikom na ikono. Podrobnosti so opisane v podpoglavju 3.4.4. V začetku je torej izpiše le koda za prikaz slike (ikone zemljevida) in katera akcija se mora izvršiti ob kliku nanjo.

Sledi vključitev vsebine modula za registracijo in prijavo, ki je opisan v podpoglavju 3.4.5.

```
<?php include("../moduli/PrijavaInRegistracija/index.php"); ?>
```

Nato se izpiše koda HTML za glavo tabele rezultatov iskanj. Njen končni izgled na spletni strani prikazuje slika 8.

Slika 8: Izgled glave tabele rezultatov iskanj na spletni strani



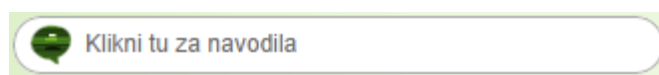
Razlogi za tak vrstni red za to so trije:

- V glavi je naslov `<h2>Spisek ekoloških kmetij po izbranih filtrih</h2>`, kar je pomembno iz optimizacijskih razlogov (SEO).
- Obiskovalec že v naprej okvirno ve, kakšne izpise rezultatov lahko pričakuje na strani.
- Ker se rezultati iskanj nalagajo asinhrono – s poizvedbo Ajax, nam ni potrebno vedno znova pošiljati glave tabele, ki se ne spreminja.

Potem se izpiše koda HTML za nogo strani (logotipi), ki je enaka na vseh podstraneh.

```
<?php include("../noga.htm"); ?>
```

Slika 9: Gumb za prikaz navodil iskalnika



Sledi izpis navodila za uporabo iskalnika, v katerera so vključene pomembne ključne besede za spletno stran (optimizacija SEO) in izpis gumba za

prikaz teh navodil. Slednja so sicer skrita in se prikažejo zgolj ob kliku na gumb (glej sliko 9).

Sledi še 5 vrstic:

```
<?php mysql_close($setupDB1->vrniPovezavo()); ?>
<script type="text/javascript" src="../js/funkcije.js"></script>
<script src="../moduli/PrijavaInRegistracija/js/prijavaInRegistracija.js"
type="text/javascript"></script>
</body>
</html>
```

Prva poskrbi, da se prekine povezavo s strežnikom MySQL, sicer bi se po nepotrebnem hranile odprte povezave.

Zadji dve vrstici naložita datoteki s funkcijami JavaScript za iskalnik in funkcijami JavaScript za modul za prijavo in registracijo. Brez teh datotek ne bi imeli interaktivne strani. V njih so shranjene vse funkcije, ki se sprožijo ob klikih na razne gumbe, ikone na spletni strani.

Dokument HTML se zaključi z `</body></html>`.

Spletni strežnik s tem ustvari celotno kodo dokumenta HTML, katerega pošlje odjemalcu, ki je dostopal do spletne strani <http://ekološke-kmetije.si>.

3.3.4 Podatkovna baza

V bazi so shranjeni vsi podatki, ki se prikažejo v filtrih iskalnika, izpisih rezultatov iskanj in na predstavitvenih straneh pridelovalcev. Hranijo se podatki o povezavah, certifikatih, krajih, ponudnikih, poštnih okrajih in regijah, podatki u uporabniških računih,...

Bazo aplikacije spletnega iskalnika, predstavitvenih strani kmetij, administrativnih strani, itd. sestavljajo tabele:

- Admin (8 stolpcev; e_mail je primarni; neodvisna tabela)
- Iskalnik (8 stolpcev; ni primarnih ključev; 5 tujih ključev)
- Kraj (4 stolpci;)
- Logotip (10 stolpcev)
- Ponudnik (22 stolpcev)
- Ponudnik_znamka (2 stolpca)
- Posebnosti (2 stolpca)
- Postni_okraj (2 stolpca)
- Predstavitev_ponudnika (4 stolpci)
- Regija (2 stolpca)
- Session (4 stolpci)
- Status (2 stolpca)
- Uporabnik (7 stolpcev)
- Zivila (1 stolpec)

Baza in podatki v tabeli uporabljajo za razvrščanje znakov pravilo utf8_slovenian_ci. Podrobnosti glede tega so navedene v podpoglavju 3.2.

Baza je v naslednjih poglavjih predstavljena skozi pregled struktur tabel, primerov vsebine tabel, opisno in preko navedb primarnih in tujih ključev relacijskega podatkovnega modela.

»Relacijski podatkovni model spada med površinske modele. Uporablja se za predstavitev baze na konceptualnem in zunanjem nivoju. Relacijska podatkovna baza se uporabniku kaže, kot množica tabel. Vsaka tabela ima čelno vrstico, ki jo imenujemo relacijska shema in podatkovne vrstice, ki jih imenujemo relacija.... V primeru, da nastopajo v relaciji funkcionalne odvisnosti, ni potrebno poznati vseh atributov, pač pa le tistih, ki zadoščajo za razlikovanje n-teric med seboj. Množici takih atributov pravimo ključ relacije. Če med atributi relacijske sheme ni funkcionalnih odvisnosti, potem tvorijo ključ vsi atributi skupa.j«. [16]

»Primarni ključ je polje, katerega podatki enolično določajo vsak zapis. Ko vemo za podatek v polju, ki je primarni ključ, lahko nedvoumno poiščemo cel zapis, ker v takem polju ne moreta obstajati dva podatka z enako vrednostjo. Polje primarnega ključa lahko vsebuje podatke, s katerimi se srečujemo vsak dan ali pa je to umetno polje, katerega vrednost določi program.« [17]

»Zunanji ali tuj ključ v neki relaciji je primarni ključ v drugi relaciji in je namenjen povezavi posameznih relacij v podatkovni bazi.« [16]

V naši podatkovni bazi se pojavljajo primarni in tuji ključi. Uporabljeno orodje phpMyAdmin omogoča, da primarne ključe dodamo s klikom na specifičen gumb pri pregledu strukture tabele. V kolikor želimo vzpostaviti tuji ključ, to naredimo po naslednjem principu:


```

ALTER TABLE Iskalnik
ADD CONSTRAINT Blagovna_znamka_fk
FOREIGN KEY (blagovna_znamka)
REFERENCES Logotip(id_logotipa)

```

Zgornja koda je poizvedba SQL, ki vstavi v tabelo `Iskalnik` nov tuji ključ z imenom `Blagovna_znamka_fk`, ki je v relaciji s primarnim ključem `id_logotipa` iz tabele `Logotip`. Slednji mora seveda biti določen. Primarni in tuji ključ morata biti istega tipa in iste dolžine, sicer se poizvedba ne izvede uspešno. Posledica takšne relacije med tabelama je, da v tabelo `Iskalnik` v polje `blagovna_znamka` ne moremo več vpisati neke vrednosti, če ta vrednost ni ena izmed vrednosti primarnih ključev tabele `Logotip`. Orodje phpMyAdmin ob vnosu novih vrstic pri poljih, ki so tuji ključi, opcije ponudi vnaprej. Primer prikazuje slika 10.

Slika 10 - phpMyAdmin - vnos nove vrstice - določanje vrednosti za polje, ki je tuji ključ

Stolpec	Vrsta	Funkcija	Null	Vrednost
iskalna_fraza	varchar(50)			
blagovna_znamka	tinyint(4)			▼
regija	tinyint(4)			0 1 2 3 4 5 6 7 8 9
postni_okraj	tinyint(4)			
status_kmetije	varchar(5)			
ponudba_kmetije	varchar(50)			
posebnosti	tinyint(4)			
Timestamp	timestamp			CURRENT_TIMESTAMP

Izvedi

Poglejmo si, kakšen je smisel uporabe primarnih in tujih ključev pri optimizaciji baze.

3.3.4.1 Indeksiranje stolpcev

Indeksi se uporabljajo za hitro iskanje vrstic s specifičnimi vrednostmi stolpcev. Brez indeksa mora MySQL začeti s prvo vrstico in potem prebrati celo tabelo, da najde relevantne vrstice. Večja, kot je tabela, več nas to stane. Če ima tabela indeks za stolpec, po katerem se išče, lahko MySQL hitro determinira iskano pozicijo vrstice, ne da bi preiskal celotno tabelo. Če ima tabela 1000 vrstic, je to vsaj 100 krat hitreje, kot če bi jo bral sekvenčno. [18]

Večina indeksov MySQL (primarni ključ, unikatni ključ, index, fulltext) je shranjenih v B-drevesu. Izjeme so, da indeksi na prostorskih podatkovnih tipih uporabljajo R-drevesa, tabele v pomnilniku pa podpirajo tudi razpršene indekse (hash indexes). [18]

MySQL uporablja indekse za naslednje operacije: [18]

- Da najde ujemanje v pogojih *WHERE*.
- Da eliminira vrstice pred tem, da bi bile podvržene ocenitvi. V primeru, da lahko MySQL izbira med več indeksi, načeloma izbere tistega, ki najde najmanjše število vrstic.
- Da vrne vrstice iz drugih tabel, ko izvaja stavke *JOIN*.

- Da najde minimalne ali maksimalne vrednosti v specifičnem stolpcu (MIN() in MAX()).
- Za razvrščanje ali grupiranje tabele v nekaterih primerih.

Za nekatere stolpce vemo, da morajo biti indeksirani, za nekatere bi bilo pa morda bolje, da jih več povežemo v en indeks. V začetku smo tabele indeksirali po tistih poljih, po katerih predvidevamo, da bo največ iskanj. Uporabili smo 1 stolpec na 1 indeks. Informacije, ki jih bomo pridobili na račun tabele `Iskalnik`, ki hrani podatke iskanj, nam bodo lahko služile za boljši izbor indeksov.

V bazi operiramo s primarnimi in tujimi ključi. Oglejmo si, kako okvirno izgleda poizvedba, ki se sproži ob uporabi iskalnika. Podrobnosti so razložene v podpoglavju 3.4.2.1.2.

```
SELECT p.*, k.ime_kraja, r.id_regije, r.ime_regije, pr.permalink
FROM Ponudnik
LEFT JOIN PredstavitevPonudnika pr ON p.id_ponudnika = pr.id_ponudnika
LEFT JOIN Ponudnik_znamka pz ON p.id_ponudnika = pz.id_ponudnika
LEFT JOIN Kraj k ON p.postna_stevilka = k.postna_stevilka
LEFT JOIN Regija r ON r.id_regije = k.id_regije
LEFT JOIN Logotip l ON l.id_logotipa = pz.id_logotipa
WHERE p.vidnost='1' AND p.certifikat = '1' AND
r.id_regije = $regije AND
k.id_okraja = $okraj AND
p.status = $status AND
p.eko_ponudba = $zivilo AND
l.id_logotipa = $blagovnaZnamka AND p.obisk_solarjev = '1' AND
```

Določanje ključev... Upoštevano je bilo, kar smo na začetku podpoglavja povedali o prednostih ključev. Ključi so bili določeni na naslednji način (kratico P smo uporabili za primarni ključ in T za tuji):

Za povezovanje preko LEFT JOIN:

- p.id_ponudnika je P, pr.id_ponudnika je T
- pz.id_ponudnika je T
- p.postna_stevilka je T, k.postna_stevilka je P
- r.id_regije je T, k.id_regije je P

Za pogoje WHERE:

- r.id_regije je P
- k.id_okraja je T
- p.status, p.eko_ponudba, p.vidnost, p.obisk_solarjev, p.certifikat niso niti primarni niti tuji ključi, zato jih lahko določimo kot indeksne ključe
- l.id_logotipa je P

V podpoglavju 3.4.2.1.2 so podane še odvisnosti od iskalne fraze. Dodatno tako dobimo še dva indeksa: l.ime_logotipa, l.organizacija.

Podroben opis ključev se nahaja v naslednjih podpoglavjih, za vsako tabelo posebej.

3.3.4.2 Tabela Admin

Je neodvisna od ostalih tabel. Uporablja se za lažji nadzor podatkov o administratorju oziroma administratorjih, katere posredujemo pridelovalcem in uporabnikom v naslednjih situacijah:

- Ko pošljemo e-sporočilo za potrditev registracije novega uporabnika
(datoteka `moduli/PrijavaInRegistracija/posljiEMailRegistracijo.php`)
- Ko pošljemo e-sporočilo za pozabljeno geslo
(datoteka `moduli/PrijavaInRegistracija/pozabljenoGeslo.php`)

- Ko prikazujemo administrativne strani uporabnikov in pridelovalcev (datoteka `pridelovalci/administracija.php`)
- Pridelovalci kmetij se lahko prijavijo v administrativni del aplikacije, kjer lahko urejajo nekatere svoje podatke in spišejo opis kmetije, ki se prikaže na predstavitveni strani kmetije. Ta del je opisan v podpoglavju 3.4.6. V primeru, da imajo pri urejanju kakšne težave, je poskrbljeno, da jim lahko administrator pomaga na način, da se sam prijavi v njihov administrativni del. Ker administrator ne pozna gesel uporabnikov (ta so v bazi kriptirana), se ob vsaki prijavi s specifičnim uporabniškim imenom preverja, ali je geslo slučajno eno od gesel administratorjev. Prijava je torej možna bodisi z ustrezno kombinacijo uporabniškega računa in njemu pripadajočega gesla bodisi z ustrezno kombinacijo uporabniškega računa in obstoječega gesla administratorja.

Sliki 11 in 12 prikazujeta podatke in strukturo tabele `Admin`.

Slika 11: Primer podatkov v tabeli Admin

ime	priimek	eMail	uplme	geslo	telefon	gsm	uradneUre
Boris	Turk	turk.boris@gmail.com	admin	??		040-215-117	od 8:00 do 16:00 od ponedeljka do petka

Slika 12: Struktura tabele Admin

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	ime	varchar(50)	utf8_slovenian_ci
2	priimek	varchar(50)	utf8_slovenian_ci
3	eMail	varchar(50)	utf8_slovenian_ci
4	uplme	varchar(50)	utf8_slovenian_ci
5	geslo	varchar(50)	utf8_slovenian_ci
6	telefon	varchar(50)	utf8_slovenian_ci
7	gsm	varchar(50)	utf8_slovenian_ci
8	uradneUre	varchar(50)	utf8_slovenian_ci

Polje eMail je primarni ključ. Tabelo Admin urejamo direktno preko phpMyAdmin.

Za dostop do podatkov tabele sta bila ustvarjena razreda `admini` in `admin`, ki se nahajata v datoteki `class.admin.php`. Njena vsebina je opisana na naslednji strani.

Zgled uporabe razreda `admin` iz datoteke `class.admin.php`:

```
<?php
$aR = new admini(); //Ustvarimo nov objekt tipa admini
$admin=$aR->vrniAdmina("admin");//Administrator z uporabniškim imenom admin
echo $admin->vrniEmail(); //Izpišemo eMail administratorja admin
?
```

```

<?php
class admini { //Razred admini
    private $admini;
    public function admini(){} //Konstruktor
    //Kot parameter posredujemo uporabniško ime administratorja.
    //Funkcija vrača objekt admin, katerega lahko uporabimo za lažji
    // dostop do ostalih podatkov o administratorju.
    public function vrniAdmina($upIme){
        $sqlAdmin="SELECT * FROM Admin WHERE upIme='$upIme'";
        if ($tempAdmin = mysql_query($sqlAdmin)){
            while($row = mysql_fetch_assoc($tempAdmin)) {
                $admin = new admin ($row["priimek"],$row["upIme"],
                                    $row["uradneUre"],$row["ime"],$row["geslo"],
                                    $row["eMail"],$row["telefon"],$row["gsm"]);

                return $admin;
            }
        } else{ echo (mysql_error ()); return -1;
        }
    }
}

//Razred admin vsebuje vse podatke, ki so na voljo o administratorju v
// podatkovni bazi v tabeli Admin
class admin {
    private $ime="";    private $priimek="";    private $geslo="";
    private $upIme="";    private $eMail="";    private $telefon="";
    private $gsm="";    private $uradneUre="";
    //Konstruktor
    public function admin($ime,$priimek,$upIme,$geslo,
                        $eMail,$telefon,$gsm,$uradneUre){
        $this->ime=$ime;    $this->priimek=$priimek;
        $this->geslo=$geslo; $this->upIme=$upIme;
        $this->eMail=$eMail; $this->telefon=$telefon;
        $this->gsm=$gsm;    $this->uradneUre=$uradneUre;
    }
    //Funkcije, ki vračajo vrednosti instance objekta admin
    public function vrniIme(){ return $this->ime; }
    public function vrniPriimek(){return $this->priimek;}
    public function vrniUpIme(){return $this->upIme;}
    public function vrniGeslo(){return $this->geslo;}
    public function vrniEMail(){return $this->eMail;}
    public function vrniTelefon(){return $this->telefon;}
    public function vrniGsm(){return $this->gsm;}
    public function vrniUradneUre(){return $this->uradneUre;}
}
?>

```

3.3.4.3 Tabela Iskalnik in indeksiranje tabel

Tabela `Iskalnik` omogoča shranjevanje informacij o vsakem iskanju s strani obiskovalca. Za pregled podatkov je pripravljen vmesnik (glej sliko 13), do katerega pridemo s klikom na povezavo »Arhiv«, ki se nahaja na strani desno v nogi.

Slika 13: Primer izpisa arhiva iskanj. Povezava do njega je v desni nogi spletne strani.



Zaenkrat še ni poskrbljeno za optimizacijo prikaza (na primer omejitev izpisa po tednih, mesecih), saj bo najprej potrebno počakati na praktične rezultate iskanj obiskovalcev iskalnika. V kolikor bi bilo iskanj veliko, bi lahko postalo izpisovanje celotnega arhiva časovno preveč požrešno. Strategija reševanja tega problema se bo razvijala naknadno.

Bistveni pomen te tabele je zaenkrat ta, da lahko ažurno spremljamo, po čem obiskovalci iščejo in kako iščejo (ali vedo na primer, da podravska regija in celjski poštni okraj ne spadata skupaj... iskalnik pri takšnem izboru ne bo vrnil rezultatov). Podatki nam bodo lahko koristili pri nadaljnji optimizaciji iskalnika in morebiti prilagoditvi filtrov.

Podatke tabele `Iskalnik` si lahko ogledamo na sliki 14, njeno strukturo pa prikazuje slika 15 na naslednji strani.

Slika 14: Primer zapisov v tabeli Iskalnik

iskalna_fraza	blagovna_znamka	regija	postni_okraj	status_kmetije	ponudba_kmetije	posebnosti	Timestamp
		0	0	0	kajmak	0	2014-09-19 14:49:38
jab&hru	1	1	2	P1	ajdova kaša	0	2014-09-17 05:45:02
jab&hru	1	1	2	P1	ajdova kaša	0	2014-09-17 05:45:02
hruške	0	1	0	0	ajda	1	2014-09-19 16:05:10
hruške	0	0	0	0	ajda	1	2014-09-19 16:05:21

Opis polj:

- `iskalna_fraza` vsebuje iskalni niz, ki ga uporabnik vpiše v iskalnik ekoloških kmetij
- `blagovna_znamka` je tuji ključ za primarni ključ `id_logotipa` iz tabele `Logotip`
- `regija` je tuji ključ za primarni ključ `id_regije` iz tabele `Regija`
- `postni_okraj` je tuji ključ za primarni ključ `id_okraja` iz tabele `Postni_okraj`
- `status_kmetije` je tuji ključ za primarni ključ `koda_statusa` iz tabele `Status`
- `ponudba_kmetije` je samo logično gledano vezan na polje `zivilo` iz tabele `Zivila`, ni pa tuji ključ, saj se teoretično gledano lahko nabor živil v tabeli `Zivila` zmanjša, če ne

obstaja noben ponudnik, ki bi živilo prideloval. Navkljub temu pa bi se v tabeli `Iskalnik` še vedno hranil arhiv o iskanju po tem živilu.

- `posebnosti` je tuji ključ za primarni ključ `id_posebnosti` iz tabele `Posebnosti`

Slika 15: Struktura tabele `Iskalnik`

#	Ime	Vrsta	Pravilo za razvrščanje znakov	Atributi
1	<code>iskalna_fraza</code>	<code>varchar(50)</code>	<code>utf8_slovenian_ci</code>	
2	<code>blagovna_znamka</code>	<code>tinyint(4)</code>		
3	<code>regija</code>	<code>tinyint(4)</code>		
4	<code>postni_okraj</code>	<code>tinyint(4)</code>		
5	<code>status_kmetije</code>	<code>varchar(50)</code>	<code>utf8_slovenian_ci</code>	
6	<code>ponudba_kmetije</code>	<code>varchar(50)</code>	<code>utf8_slovenian_ci</code>	
7	<code>posebnosti</code>	<code>varchar(100)</code>	<code>utf8_slovenian_ci</code>	
8	<code>Timestamp</code>	<code>timestamp</code>		<code>on update CURRENT_TIMESTAMP</code>

Vse vrednosti polj, razen polja `iskalna_fraza`, so vedno določene – niso prazne oziroma `Null`. Polje `Timestamp` se določi samodejno in določa čas, kdaj je obiskovalec iskal.

V tabelo `Iskalnik` piše skripta `izpisiRezultatovIskanj.php`, ki je podrobno opisana v podpoglavju 3.4.2.1.2.

3.3.4.4 Tabela Kraj

Tabela služi kot povezava med regijo, poštним okrajem in poštno številko. V tabeli se nahaja 479 zapisov. Ko dodamo nov zapis v tabelo `Ponudnik`, notri vpišemo naslov kmetije (ulico oziroma vas) ter poštno številko. Kraja, regije in poštnega okraja ne vpisujemo. Razloga za to sta dva:

- Doslej nikjer nismo zasledili vira podatkov o ekoloških pridelovalcih, ki bi vseboval vse te podatke. Tako nam vedno en del manjka. Načeloma pa sta vedno prisotna poštna številka in naslov.
- V načrtu imamo nadgrajevati opcije iskalnika ekoloških kmetij. V kolikor bomo na spletu zasledili nove sezname (na primer seznam ekoloških kmetij, ki nudijo prebivališče, možnost zaposlitve, ali ki sprejemajo vajence, ipd.), nam bo v interesu dopolniti filter »Posebnosti« (glej sliko 16) z novimi opcijami. V takšnem slučaju bi seveda bilo potrebno posodobiti tudi tabelo `Ponudniki`. Denimo, da bi dodali nov stolpec `sprejemajo_vajence`. Ta bi bil tipa `boolean`. Nato bi morali vrednost tega polja nastaviti na 1 za vse ponudnike, ki bi sprejemali vajence. Nazivi krajev z referenčnih podatkov se morda zaradi človeških napak na referenčnih straneh ne bi ujemali z našimi. Posledično bi dobili v bazi spisek podvojenih ali neobstoječih krajev.

Slika 16: Filter »Posebnosti« se bo v prihodnje dopolnjeval

Podatki o krajih in poštne številke so pridobljeni s strani Pošte Slovenije^w. Številke okrajev so pridobljene na osnovi kopiranja prve številke poštne številke v stolpec `id_okraja`. Vzpostaviti povezavo med regijami in kraji je bila zamudnejša naloga. Želeli bi, da bi lahko kakšne preslikovalne tabele med kraji in regijami našli bodisi na Pošti Slovenije bodisi v Telefonskem imeniku bodisi na Wikipediji. Vendar o teh ne duha ne sluha. Znano je, katere občine spadajo v katero regijo^x. Nato je bilo za 212 občin pregledano, katera naselja jim pripadajo. Za vsako naselje pa je znano, katero pošto številko ima^y.

Podatke in strukturo tabele `Kraj` prikazujeta sliki 17 in 18.

Slika 17: Primer zapisov v tabeli `Kraj`

<code>postna_stevilka</code>	<code>ime_kraja</code>	<code>id_okraja</code>	<code>id_regije</code>
1000	Ljubljana	1	8
1001	Ljubljana – P.P.	1	8
1122	Ljubljana	1	8
1210	Ljubljana – Šentvid	1	8
1211	Ljubljana – Šmartno	1	8
1215	Medvode	1	8

Slika 18: Struktura tabele `Kraj`

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	<code>postna_stevilka</code>	<code>int(11)</code>	
2	<code>ime_kraja</code>	<code>varchar(100)</code>	<code>utf8_slovenian_ci</code>
3	<code>id_okraja</code>	<code>varchar(100)</code>	<code>utf8_slovenian_ci</code>
4	<code>id_regije</code>	<code>int(11)</code>	

Opis polj:

- `postna_stevilka` je primarni ključ.
- `ime_kraja` se lahko ponavlja, vendar gre za različne poštne številke (npr. 2000 Maribor, 2101 Maribor, 2103 Maribor).
- `id_regije` je tuji ključ za primarni ključ `id_regije` iz tabele `Regija`.

3.3.4.5 Tabela `Logotip`

Tabela služi beleženju informacij o organizacijah, ki so povezane s certifikati in ekološkimi kmetijami.

Zapisi, ki imajo v stolpcu `bool_blagovna_znamka` vrednost 1, so blagovne znamke za pridelavo ekoloških živil, registrirane v Sloveniji. Logotipi Biodar, Demeter in Ekodar so logotipi blagovnih znamk.

Logotipi Bio Slovenija, SI IKC ECO in Bureau Veritas so logotipi organov, ki izdajajo certifikate.

Logotipi Ekološko kmetijstvo EU, Zeleni list EU in Ekološki Nacionalni so logotipi, ki niso niti blagovne znamke niti organi za certificiranje. Imajo pa podoben pomen, kot logotipi

^w Pošta Slovenije objavlja spisek poštne številke doma na strani www.posta.si/postne-stevilke-doma

^x Wikipedija: Statistične regije Slovenije – Občine po statističnih regijah:
http://sl.wikipedia.org/wiki/Statisti%C4%8Dne_regije_Slovenije

^y Wikipedija: Seznam naselij v Sloveniji: http://sl.wikipedia.org/wiki/Seznam_naselij_v_Sloveniji

blagovnih znamk – potrjujejo, da je izdelek ekološki. Logotip Ekološko kmetijstvo EU je v opuščanju, ker ga je nadomestil logotip Zeleni list EU.

Nekateri stolpci tabele so namenjeni zgolj boljši preglednosti nad posameznimi institucijami in so v oporo administratorju.

Pri generiranju kode za filter »Blagovna znamka« (prisoten na spletni strani) se vrednosti lokacij slik naložijo iz stolpca `ime_slike`.

Podatke in strukturo tabele `Logotip` prikazujeta sliki 19 in 20.

Slika 19: Zgled zapisov v tabeli Logotip

id_logotipa	ime_logotipa	tip_slike	ime_slike	organizacija	kratica_organizacije	bool_blagovna_znamka	opomba	bool_certifikacijski_organ	daljsi_opis
0						0		0	
1	Biodar	image/png	biodar.png	Zveza združenj ekoloških kmetov Slovenije Biodar		1	Zasebna blagovna znamka za ekološko pridelavo.	0	Zveza združenj ekoloških kmetov Slovenije (ZZEKS)
2	Bio Slovenija	image/png	bio-slovenija.png	Inštitut za kontrolo in certifikacijo v kmetijstvu	KON-CERT	0	Njihova koda: SI-EKO-001 (prej SI-01-EKO)	1	
3	Ekološko kmetijstvo EU	image/png	ekolosko-kmetijstvo.png	Evropska unija		0	Predhodni evropski znak za ekološko kmetijstvo	0	Predhodni evropski znak za ekološko kmetijstvo. Se...

Slika 20: Struktura tabele Logotip

#	Ime	Vrsta	Pravilo za razvrščanje znakov	Atributi
1	<code>id_logotipa</code>	int(11)		
2	<code>ime_logotipa</code>	varchar(100)	utf8_slovenian_ci	
3	<code>tip_slike</code>	varchar(25)	utf8_slovenian_ci	
4	<code>ime_slike</code>	varchar(50)	utf8_slovenian_ci	
5	<code>organizacija</code>	varchar(100)	utf8_slovenian_ci	
6	<code>bool_blagovna_znamka</code>	tinyint(1)		
7	<code>kratica_organizacije</code>	varchar(20)	latin1_swedish_ci	
8	<code>opomba</code>	varchar(200)	utf8_slovenian_ci	
9	<code>bool_certifikacijski_organ</code>	tinyint(1)		
10	<code>daljsi_opis</code>	varchar(2000)	utf8_slovenian_ci	

`id_logotipa` je primarni ključ. Nanj je vezan filter za iskanje po blagovni znamki. Filter »Iskalna fraza« pa je vezan na stolpca `ime_logotipa` in `organizacija`, zato ju določimo kot indeksna ključa.

3.3.4.6 Tabela Ponudnik

Zaradi prevelike velikosti primera izpisa tabele `Ponudnik` ne podajamo. Slednja hrani bistvene podatke o pridelovalcih ekoloških živil. Ponudnik lahko obstaja v tabeli pod pogojem, da je naveden na spisku ekoloških kmetovalcev. Strukturo tabele prikazuje slika 21.

Ministrstvo za kmetijstvo in okolje (MKO) objavi seznam certificiranih kmetij 1x na leto. 1 leto je med drugim veljaven tudi ekološki certifikat. Da bi bili podatki v naši bazi in na pripravljenem zemljevidu Google Maps Lite ažurni, se mora podatke posodobiti 1x letno. Upamo, da bomo z MKO lahko sodelovali in vzpostavili sistem, kjer bo posodabljanje podatkov lahko potekalo avtomatizirano.

Slika 21: Struktura tabele Ponudnik

#	Ime	Vrsta	Pravilo za razvrščanje znakov	Atributi	Null	Privzeto	Dodatno
1	<code>id_ponudnika</code>	<code>int(11)</code>			Ne	Brez	<code>AUTO_INCREMENT</code>
2	<code>kontaktna_oseba</code>	<code>varchar(200)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
3	<code>naziv_ponudnika</code>	<code>varchar(200)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
4	<code>telefon</code>	<code>varchar(100)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
5	<code>gsm</code>	<code>varchar(100)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
6	<code>e_mail</code>	<code>varchar(100)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
7	<code>spletna_stran</code>	<code>varchar(200)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
8	<code>referenca</code>	<code>varchar(200)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
9	<code>referencni_url</code>	<code>varchar(100)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
10	<code>naslov</code>	<code>varchar(200)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
11	<code>postna_stevilka</code>	<code>int(11)</code>			Ne	Brez	
12	<code>status</code>	<code>varchar(10)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
13	<code>zdruzenje</code>	<code>varchar(200)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
14	<code>bool_blagovna_znamka</code>	<code>tinyint(1)</code>			Ne	Brez	
15	<code>eko_ponudba</code>	<code>varchar(1000)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
16	<code>opis</code>	<code>varchar(1000)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
17	<code>kraj_cas_prodaje</code>	<code>varchar(300)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
18	<code>certifikat</code>	<code>varchar(50)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
19	<code>up_ime</code>	<code>varchar(50)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
20	<code>geslo</code>	<code>varchar(50)</code>	<code>utf8_slovenian_ci</code>		Ne	Brez	
21	<code>vidnost</code>	<code>tinyint(1)</code>			Ne	1	
22	<code>obisk_solarjev</code>	<code>tinyint(1)</code>			Ne	Brez	

Ključni:

- Primarni ključ je `id_ponudnika`.
- `postna_stevilka` je tuji ključ za primarni ključ `postna_stevilka` iz tabele `Kraj`.
- Indeksni ključji so določeni za stolpce: `obisk_solarjev`, `eko_ponudba`, `naslov`, `postna_st`, `kontaktna_oseba`, `naziv_ponudnika`, `telefon`, `gsm`, `e_mail`, `zdruzenje`. Kateri izmed teh indeksnih ključev je najprimernejši, se še ne ve. To bomo lahko razpoznali po analizi rezultatov iskanj. Več o tem piše v podpoglavju 3.3.4.3.

Stolpec `bool blagovna znamka` služi hitremu preverjanju ali se kmetija navezuje na specifično blagovno znamko. Teoretično se lahko navezuje na več blagovnih znamk. Lahko bi imela celo lastno blagovno znamko. Za te namene je bila ustvarjena tabela `Ponudnik_znamka` (glej podpoglavje 3.3.4.7).

Stolpec `opis` ni v uporabi. Nadomešča ga stolpec `predstavitev` v tabeli `Predstavitev_ponudnika`. To je narejeno iz praktičnih razlogov - boljšega pregleda nad predstavitvenimi stranmi.

Stolpec `certifikat` je namenjen temu, da se v primeru, če kmetiji zapade certifikat, ali pa če dobimo obvestilo, da ji je bil certifikat odvzet (inšpekcijske službe), vrednost nastavi na 0. Povezava med kmetijo in certifikatom je vzpostavljena na način, ki je podrobno opisan na koncu podpoglavja 3.4.6.1.

Vrednosti stolpcev `vidnost` in `obisk solarjev` lahko uporabnik določa preko administrativnega vmesnika, predstavljenega v podpoglavju 3.4.6.2. `Obisk solarjev` je opcija iskalnega filtra Posebnosti. Stolpec `vidnost` določa ali naj bodo podatki o kmetiji vidni javnosti.

Način, kako v začetku vsem pridelovalcem dodelimo naključna gesla:

```
UPDATE Ponudnik
SET geslo = CONCAT (
substring( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789' ,
rand(@seed:=10)*54+1,1) ,
substring( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789' ,
rand(@seed:=round(rand(@seed)*12345))*54+1,1) ,
substring( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789' ,
rand(@seed:=round(rand(@seed)*23456))*54+1,1) ,
substring( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789' ,
rand(@seed:=round(rand(@seed)*34567))*54+1,1) ,
substring( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789' ,
rand(@seed:=round(rand(@seed)*45678))*54+1,1) ,
substring( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789' ,
rand(@seed:=round(rand(@seed)*56789))*54+1,1) ,
substring( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789' ,
rand(@seed:=round(rand(@seed)*67890))*54+1,1) ,
substring( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789' ,
rand(@seed:=round(rand(@seed)*78901))*54+1,1) )
```

3.3.4.7 Tabela Ponudnik_znamka

Tabela je vmesni člen med tabelo Logotip in tabelo Ponudnik. Vsak pridelovalec ekoloških živil ima lahko nič ali več blagovnih znamk. Sliki 22 in 23 predstavljata njeno strukturo in podatke.

Slika 22: Struktura tabele Ponudnik_znamka

#	Ime
1	id_ponudnika
2	id_logotipa

- `id_ponudnika` je tuji ključ za primarni ključ tabele `Ponudnik`
- `id_logotipa` je tuji ključ za primarni ključ tabele `Logotip`

Slika 23: Primer zapisov tabele Ponudnik_znamka

id_ponudnika	id_logotipa
226	5
250	5
251	5
252	5
276	5
277	5

3.3.4.8 Tabela Posebnosti

Uporablja se za shrambo opcij, ki so na voljo za filter »Posebnosti« na spletni strani iskalnika:

Slike 24, 25, 26 po vrsti prikazujejo izgled filtra, podatke v tabeli, strukturo tabele.

Slika 24: Opcije filtra Posebnosti na spletni strani

Slika 25: Zapisi v tabeli Posebnosti

id_posebnosti	posebnost
1	Sprejemajo obiska šolarjev

id_posebnosti je primarni ključ

Slika 26: Struktura tabele Posebnosti

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	id_posebnosti	int(11)	
2	posebnost	varchar(100)	utf8_slovenian_ci

3.3.4.9 Tabela Postni_okraj

Uporablja se za shrambo opcij, ki so na voljo za filter »Poštni okraj«.

Slike 27, 28, 29 po vrsti prikazujejo strukturo tabele, izgled filtra na spletni strani, podatke v tabeli Postni_okraj.

Slika 27: Zapisi v tabeli Postni_okraj

id_okraja	ime_okraja
1	Ljubljanski
2	Mariborski
3	Celjski
4	Kranjski
5	Novogoriški
6	Koprski
8	Novomeški
9	Murskosoboški

Slika 27: Struktura tabele Postni_okraj

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	id_okraja	tinyint(4)	
2	ime_okraja	varchar(100)	utf8_slovenian_ci

id_okraja je primarni ključ

3.3.4.10 Tabela Predstavitev_ponudnika

Primere zapisov iz tabele in njeno strukturo prikazujeta sliki 28 in 29. V tabeli se hranijo:

- Relativni stalni naslovi predstavitev strani ponudnikov v polju `permalink`.
- V polju `predstavitev` je morebitni opis kmetije, ki ga vsak pridelovalec določi sam. Polje `staraPredstavitev` še ni v uporabi. Ideja je, da bo služilo kot arhiv starejših zapisov, za potrebe funkcije razveljavitve spremembe predstavitev.

Slika 28: Primer zapisov v tabeli Predstavitev_ponudnika

id_ponudnika	permalink	predstavitev	staraPredstavitev
1	kmetija-ronkaldo	<h2 style="padding: 5px 5px 6px 13px; font-family:...	
2	ambrozic-lidija		
3	pri-mohru		

Slika 29: Struktura tabele Predstavitev_ponudnika

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	id_ponudnika	int(11)	
2	permalink	varchar(50)	utf8_slovenian_ci
3	predstavitev	varchar(10000)	utf8_slovenian_ci
4	staraPredstavitev	varchar(10000)	utf8_slovenian_ci

Tabela `PredstavitevPonudnika` je bila zgrajena tako, da se je v stolpec `permalink` prekopiralo podatke bodisi s polja `kontaktna_oseba` bodisi s polja `naziv_ponudnika` (odvisno od tega, ali je bilo polje `nazi_ponudnika` prazno). Pri tem se je vse velike črke nadomestilo z malimi, vse znake »č,ć,š,ž« z znaki »c,c,s,z« in vse presledke z znakom »-«:

```
INSERT INTO PredstavitevPonudnika (id_ponudnika)
SELECT id_ponudnika
FROM Ponudnik;

UPDATE PredstavitevPonudnika pp
left join Ponudnik p on p.id_ponudnika = pp.id_ponudnika
SET staraPredstavitev =
replace(
  replace(
    replace(
      replace(
        lower( if(naziv_ponudnika <> '', replace(naziv_ponudnika,' ','-'),
          replace(kontaktna_oseba,' ','-'))),
        ' ','-'),
        'č','c'),
        'ć','c'),
        'š','s'),
        'ž','z')
```

3.3.4.11 Tabela Regija

Uporablja se za shrambo opcij, ki jih ponuja filter »Regija« na spletni strani iskalnika: Podatke, ki jih hrani tabela, izgled filtra na spletni strani in strukturo tabele prikazujejo slike 30, 31, 32.

Slika 30: Zapisi v tabeli Regija

id_regije	ime_regije
1	Pomurska
2	Podravska
3	Koroška
4	Savinjska
5	Zasavska
6	Spodnjeposavska
7	Jugovzhodna
8	Osrednjeslovenska
9	Gorenjska
10	Notranjsko - kraška
11	Goriška
12	Obalno - kraška

Slika 31: Opcije filtra Regija na spletni strani iskalnika

Slika 32: Struktura tabele Regija

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	<u>id_regije</u>	tinyint(4)	
2	ime_regije	varchar(100)	utf8_slovenian_ci

id_regije je primarni ključ

3.3.4.12 Tabela session

Tabela se uporablja za potrebe shranjevanja sej (ang. session) ob prijavi uporabnika v administrativni del aplikacije. Princip delovanja sej je razložen v podpoglavju 3.4.6.2. ID je primarni ključ. Podatke in strukturo tabele session prikazujeta sliki 33 in 34.

Slika 33: Primer zapisov v tabeli session

ID	SessionID	Data	DateTouched
2	su8n7kaqv3dfoahu911obut6g6	EkoKmetijeUporabnik s:7:"adamir2";EkoKmetijeGeslo ...	1410921769
8	ehein3s7us48duol3svfveges5	EkoKmetijeUporabnik s:7:"adamir2";EkoKmetijeGeslo ...	1411104066
9	f21su0e5esr7g3k9631a5ra3l3	EkoKmetijeUporabnik s:7:"adamir2";EkoKmetijeGeslo ...	1411105674
10	0fdedjism31de9iem1c9ik85e1	EkoKmetijeUporabnik s:7:"adamir2";EkoKmetijeGeslo ...	1411131200

Slika 34: Struktura tabele session

#	Ime	Vrsta	Pravilo za razvrščanje znakov	Atributi	Null	Privzeto	Dodatno
1	<u>ID</u>	int(11)			Ne	Brez	AUTO_INCREMENT
2	SessionID	char(26)	utf8_slovenian_ci		Da	NULL	
3	Data	text	utf8_slovenian_ci		Da	NULL	
4	DateTouched	int(11)			Da	NULL	

3.3.4.13 Tabela Status

Uporablja se za shrambo opcij, ki jih ponuja filter »Status kmetije«. To so splošni statusi. Pri blagovni znamki Demeter v resnici uporabljajo več statusov. V prihodnosti bo potrebno o tem pridobiti

več informacij. Statusi določajo ali kmetija že ima certifikat ali je še v postopku preusmeritve, ki traja več let.. Izgled filtra »Status kmetije« na spletni strani, strukturo tabele Status in zapise v njej prikazujejo slike 35, 36, 37.

Slika 35: Opcije filtra »Status kmetije« na spletni strani iskalnika

Slika 36: Struktura tabele Status

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	<u>koda_statusa</u>	varchar(5)	utf8_slovenian_ci
2	<u>opis_statusa</u>	varchar(100)	utf8_slovenian_ci

koda_statusa je primarni ključ.

Slika 37: Zapisi v tabeli Status

koda_statusa	opis_statusa
P1	P1 - prvo leto preusmeritve
P2	P2 - drugo leto preusmeritve
P3	P3 - tretje leto preusmeritve
E	E - ekološka kmetija

3.3.4.14 Tabela Uporabnik

Podatke tabele Uporabnik in njeno strukturo prikazujeta sliki 38, 39. Vanjo se doda nov zapis ob uspešni registraciji uporabnika. Vsi uporabniki morajo biti v bazi že prej, saj za njih jamči MKO. Problem je zgolj v verifikaciji. Podroben opis tega je podan v podpoglavju 3.4.5.

Slika 38: Primer zapisov v tabeli Uporabnik

ime	priimek	uplme	geslo	eMail	aktiviran	telefon	validacijska_koda
Test	Test	test	b59d67d45a78030e094e04f95fbae35b	test1.boris@gmail.com	1	051000000	
Boris	Turk	admin	c3284d0f94606de1fd2af172aba15bf3	testAdmin@gmai.com	0	040215117	
Marjan	Božič	marbozic	adc5eca334f96c65d3588af7eef377e2	testMarjan@gmail.com	0	040000000	
Zvone	Kastelic	zvonkaste	1dde8f3ba88eb07766261d698ca7f516	testZvone@gmai.com	0	040000000	

Slika 39: Struktura tabele Uporabnik

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	<u>ime</u>	varchar(100)	utf8_slovenian_ci
2	<u>priimek</u>	varchar(100)	utf8_slovenian_ci
3	<u>uplme</u>	varchar(20)	utf8_slovenian_ci
4	<u>geslo</u>	varchar(128)	utf8_slovenian_ci
5	<u>eMail</u>	varchar(100)	utf8_slovenian_ci
6	aktiviran	tinyint(1)	
7	telefon	varchar(20)	utf8_slovenian_ci
8	validacijska_koda	varchar(10)	utf8_slovenian_ci

eMail je primarni ključ

Vsi podatki, razen telefona in validacijske kode, so vedno prisotni. To pogojuje forma za registracijo. Vrednost polja aktiviran v začetku je 0. Za tem, ko uporabnik potrdi registracijo, se postavi na 1. Pomen validacijske kode je pojasnjen v podpoglavju 3.4.5 .

3.3.4.15 Tabela Zivila

V tabeli `Zivila` so vsa živila, ki jih ponujajo kmetije v naši bazi. Primere podatkov in strukturo tabele prikazujeta sliki 40, 41. Za enkrat nam še ni znano, ali je spisek živil pri kakšni instituciji za certificiranje v naprej znan.

Slika 40: Primer zapisov v tabeli Zivila

živilo
ajda
ajdova kaša
ajdova moka
ameriške borovnice
aronija

Slika 41: Struktura tabele Zivila

#	Ime	Vrsta	Pravilo za razvrščanje znakov
1	živilo	varchar(100)	utf8_slovenian_ci

Ko dodamo v tabelo `Ponudnik` novo kmetijo, moramo posodobiti tabelo `Zivila` s podatki iz polja `eko_ponudba`, ki je v tabeli `Ponudnik`. V njem so z vejicami ločena živila, za ka katere kmetija poseduje certifikat. Spodaj je opisana skripta, ki opravi to delo.

```
<?php
header('Content-Type: text/html; charset=utf-8');
//Vključimo datoteko za nastavitve strežnika, baze, jezika
include("../..../class.StreznikInBaza.php");
$setupDB1 = vrniKonfiguriranoBazo(); //Ustvarimo instanco razreda StreznikInBaza.
$vsaZivila = array(); //To bo naš seznam vseh živil vseh pridelovalcev ekoloških živil.
//V tabeli Ponudnik je v stolpcu eko_ponudba za vsakega ponudnika (kmetijo)
// podan spisek živil, ki jih prideluje, oziroma ima za njih ekološki certifikat.
// Pridobiti želimo seznam vseh živil, na način, da se živila ne bodo ponavljala.
$query="SELECT DISTINCT eko_ponudba FROM `Ponudnik`";
if ($ponudniki = mysql_query($query)){ //Poženemo poizvedbo sql, zapisano v $query.
    //Rezultat $ponudniki je seznam vrstic. Vsaka vrstica hrani seznam živil.
    while($row = mysql_fetch_assoc($ponudniki)) {
        $eko_ponudba = $row["eko_ponudba"];
        //Besede znotraj ene vrstice napolnimo v seznam z ukazom explode.
        $stringArray = explode(' ', $eko_ponudba);
        foreach($stringArray as $zivilo) {
            //lahko se zgodi, da je isto živilo napisano na različne načine
            // (velike/male črke), zato besedo pretvorimo v male črke.
            $zivilo = mb_strtolower ($zivilo, 'UTF-8');
            //Živilo ima lahko na začetku ali na koncu presledke, npr.
            // " ajdova kaša"... takšnih presledkov se znebimo, ostale pustimo.
            $zivilo = trim($zivilo);
            //V kolikor živilo še ne obstaja v seznamu, ga dodamo.
            if ((! in_array($zivilo, $vsaZivila, true)) and ($zivilo <> '')) {
                array_push($vsaZivila, $zivilo);
            } //Konec za for in konec za while.
        }
    } else{ echo (mysql_error ()); } //Izpis morebitne napake.
natsort($vsaZivila); //Razvrstimo živila po abecednem vrstnem redu.
//Pobrišemo vsebino tabele Zivila.
if (! mysql_query("DELETE FROM Zivila")){ echo (mysql_error ()); }
foreach($vsaZivila as $zivilo){ //Napolnimo tabelo Zivila z novimi vrednostmi.
    if (! mysql_query("INSERT INTO Zivila(zivilo) VALUES ('".$zivilo."')")) {
        echo (mysql_error ()); } else{ echo($zivilo.","); } //Izpis napake ali pa živila.
    }
}
mysql_close($setupDB1->vrniPovezavo()); //Zapremo povezavo s strežnikom.
?>
```

3.4 Spletna stran

V tem podpoglavju je predstavljeno, kako lahko obiskovalec uporablja spletno stran; kaj mu stran nudi; kako deluje spletni iskalnik; kateri podatki se prikažejo v rezultatih iskanj; kako lahko prikažemo zemljevid; kako se certificirani uporabniki prijavijo, registrirajo, pošljejo zahtevo za pozabljeno geslo; kako deluje mehanizem za prikaz predstavitvenih strani kmetij; kaj lahko aktivirani uporabniki urejajo v administrativnem delu aplikacije in kako lahko spišejo svoje oblikovano besedilo, ki se prikaže na predstavitveni strani specifične kmetije.

3.4.1 Uporaba iskalnika

Obiskovalcu aplikacija ponuja kratka navodila za uporabo iskalnika, katera je pomembno omeniti, da razumemo način delovanja iskalnika. Prikažejo se v obliki, kot kaže slika 42.

Slika 42: Prikaz navodil za iskanje na spletni strani iskalnika



Iskalnik zna obravnavati tudi »napake«. Če na primer za iskalno frazo vnesemo niz

»,, , jab , hru & sir, pira,& sir,moka & «,
to iskalnik prevede v
»jab ALI (hru IN sir) ALI pira ALI sir ALI moka«

Podrobnosti mehanizma, ki to omogoča, so razložene v podpoglavju 3.4.2.1.2.

3.4.2 Delovanje iskalnika - spisek ekoloških kmetij glede na izbrane filtre

Obiskovalec spletnega iskalnika ekoloških kmetij lahko preiskuje našo bazo bodisi z vpisom iskalne fraze bodisi z izbiro opcij v specifičnih filtrih. Obe možnosti delujeta v okviru spletne forme `<form>`. Slika 43 prikazuje, kakšna izgleda forma na spletni strani.

Iskanje po iskalni frazi podpira operatorja IN in ALI. IN je predstavljen z vejico, ALI pa z znakom &. Vmes so lahko presledki. Podrobnejša obravnava tega mehanizma je opisana v podpoglavju 3.4.2.1.2.

Slika 43: Forma za iskanje po ekoloških kmetijah, kakršno vidi uporabnik

The image shows a web form for searching organic farms. It has a search bar at the top with the placeholder text 'Vpišite besede, povezane s kmetijo'. Below the search bar is a section titled 'Dodatne opcije' (Additional options) which contains several dropdown menus: 'Blagovna znamka' (Brand), 'Regija' (Region), 'Poštni okraj' (Postal code), 'Status kmetije' (Farm status), 'Ponudba kmetije' (Farm offer), and 'Posebnosti' (Specialties). All dropdowns currently show 'Nedoločeno' (Undefined). At the bottom of the form is a green button labeled 'Izpiši' (Print) and a link that says 'Klikni tu za navodila' (Click here for instructions).

Spodaj si bomo ogledali del kode znotraj forme, iz katere so odstranjeni bloki, ki za delovanje forme nimajo pomena. V mislih imejte, da ne operiramo s standardnim HTML `<select>` elementom, saj uporabljamo posebno knjižnico, ki omogoča lepši izpis in dodatne funkcionalnosti tega elementa (glej podpoglavje 3.4.7).

```
<form id="ekoPonudnikiForm" method="post" onsubmit=""
  action="javascript:processForm();" >
  <input type="text" name="iskalnaFraza"
    onclick=" $('form#ekoPonudnikiForm .error').html ('');"
    id="iskalnaFraza" value="" class="txt requiredField"
    placeholder="Vpišite besede, povezane s kmetijo" />
  <select name="postni-okraj" class="txt-jumpmenu" id="okraj"
    onchange=" $(' '#okraj') .blur();" >
    <option value="0" data-description="Nedoločeno" ></option>
    <option value="1" data-description="Ljubljanski" ></option>
    ... itd ...
    <option value="9" data-description="Murskosoboški"></option>
  </select>
  <button id="poisciKmetije" name="submit" type="submit"
    class="subbutton zeleniGumb" style="width:100%;">Izpiši
  </button>
</form>
```

Približno takšna bi izgledala naša forma, če bi uporabili le filtra »Iskalna fraza« in »Poštni okraj«. Ker je gumb »Izpiši« tipa `submit`, se po njegovem kliku kliče funkcija, določena v parametru `action` forme. To je funkcija `processForm()`.

Postopek iskanja lahko strnemo v tri faze:

- Klic funkcije `processForm()` in Ajax klic skripte `izpisRezultatovIskanj.php` na (oddaljenem) strežniku.
- Strežnik interpretira ukaze znotraj slednje skripte, vstavi nov zapis v tabelo `Iskalnik`, nato pa v izhodno datoteko izpiše tabelo rezultatov iskanj. Ko konča, pošlje datoteko odjemalcu.
- Odjemalec preko funkcije `stateChangedPonudniki()` obdela odgovor (niz, dobljen s strani strežnika) in rezultat prepiše v telo specifičnega elementa HTML.

3.4.2.1.1 Funkcija `processForm()`

Funkcija v bistvu le preda delo dvem drugim funkcijam:

- `PoizvedbaZaIzpisPonudnikov();`
- `NastaviPreprecevanjeZapuscjanjaStrani();`

AJAX (Asynchronous Javascript And Xml) je tehnika asinhronnega povezovanja med odjemalcem in strežnikom. Ključni objekt takšnega načina povezovanja je objekt `XMLHttpRequest`. Funkcija `poizvedbaZaIzpisPonudnikov()` ustvari objekt `XMLHttpRequest`, ki deluje kot rokovalnik. Uporablja se za to, da si spletni stražnik in odjemalec izmenjata podatke v ozadju, medtem ko obiskovalec spletne strani nadaljuje z delom na tej strani.

Po naših izkušnjah drži, kar navajajo [19], da so objekt `XMLHttpRequest` sanje razvijalca, ker:

- *»Posodobi spletno stran brez ponovnega nalaganja strani«*
- *»Zahteva podatke od strežnika za tem, ko se stran naloži«*
- *»Prejme podatke s strani spletnega strežnika zatem ko se stran naloži«*
- *»Pošlje strežniku podatke v ozadju«*

Na naslednji strani je funkcija `PoizvedbaZaIzpisPonudnikov()` opisana v komentarjih kode.


```

function PoizvedbaZaIzpisPonudnikov(){
    //ekoPonudnikiForm je forma, iz katere želimo izvedeti, kakšne iskalne
    // opcije je izbral obiskovalec.
    //Ob klicu php skripte izpisRezultatovIskanj.php posredujemo spletnem
    // strežniku te vrednosti kot parametre z uporabo metode GET.
    var ekoPonudnikiForm = document.getElementById('ekoPonudnikiForm');

    //Ustvarimo objekt XMLHttpRequest, opisan na prejšnji strani.
    xmlhttp=GetXmlHttpRequest();
    if (xmlhttp==null) {
        alert ("Tvoj brskalnik ne podpira XMLHTTP!");    return;
    }
    //baseDir je relativni naslov in ga nastavimo v datoteki spletne strani.
    var url=baseDir+"izpisiIskalnika/izpisRezultatovIskanj.php";

    //Določimo URL in parametre iskalnaFraza, blagovnaZnamka,...
    //V začetku poglavja je omenjeno, da iskalnik podpira operaciji IN in
    //ALI. Ker so parametri ločeni z istim znakom, kot iskalna fraza
    //(hruške&jabolka), je treba znak & zakodirati, sicer bi poslali napačne
    //parametre. To stori funkcija encodeURIComponent.Potem lahko parameter
    //vsebuje tudi znak &, ki služi kot logična operacija IN.

    url="?iskalnaFraza="+encodeURIComponent(ekoPonudnikiForm.iskalnaFraza.value);
    url=url+"&blagovnaZnamka="+ekoPonudnikiForm.blagovnaZnamka.value;
    url=url+"&regija="+ekoPonudnikiForm.regija.value;
    url=url+"&okraj="+ekoPonudnikiForm.okraj.value;
    url=url+"&status="+ekoPonudnikiForm.status.value;
    url=url+"&zivilo="+ekoPonudnikiForm.zivilo.value;
    url=url+"&posebnosti="+ekoPonudnikiForm.posebnosti.value;
    //Določimo katera funkcija se bo sprožila,
    // ko bo objekt xmlhttp od strežnika prejel odgovor
    xmlhttp.onreadystatechange=stateChangedPonudniki;
    //Objekt xmlhttp postane odjemalec in dostopa do naslova URL z metodo
    //GET, pri kateri so parametri posredovani v naslovu, ločeni z znakom &
    document.getElementById("stevilo-zadetkov-iskanj").innerHTML="Iskanje
    poteka v ozadju..."; //Če je strežnik počasen, rezultati ne bodo prikazani
    takoj. Upoorabnika obvestimo, da iskanja še ni konec in da poteka v ozadju.
    xmlhttp.open("GET",url,true);
    xmlhttp.send(null);
    return;
}

//Funkcija, ki se kliče znotraj funkcije poizvedbaZaIzpisPonudnikov()
function GetXmlHttpRequest(){
    if (window.XMLHttpRequest) { // koda za IE7+, Firefox, Chrome, Opera, Safari
        return new XMLHttpRequest();
    }
    if (window.ActiveXObject) { // koda za IE6, IE5
        return new ActiveXObject("Microsoft.XMLHTTP");
    }
    return null;
}

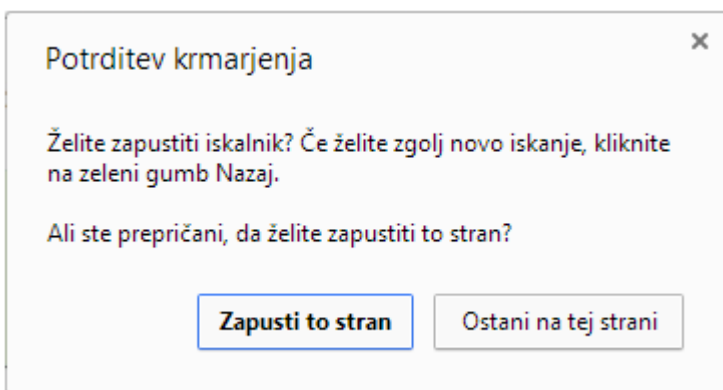
```

Medtem ko objekt `xmlhttp` v ozadju komunicira s strežnikom, pa JavaScript nadaljuje svoje delo – izvesti je potrebno še funkcijo `nastaviPreprecevanjeZapuscjanjaStrani()`. Gre za zelo priročno funkcijo, ki prepreči, da bi uporabnik nehoti zapustil stran s klikom na gumb »Nazaj«, ki je del brskalnika. Aplikacija je zasnovana tako, da se ob kliku na gumb »Išči« forma za vnos skrije. Namen tega je, da ostane več prostora za rezultate iskanja. Uporabnik lahko zaradi tega misli, da je prišel na drugo stran. Ko želi ponovno iskati, mora opaziti zelo pomemben zeleni gumb »Nazaj«, ki je prikazan na sliki 44.

Slika 44: Gumb »Nazaj« na spletni strani iskalnika



Slika 45: Brskalniskovo okno opozori uporabnika na morebitno neželeno zapušcanje strani



Če ga ne opazi in klikne standardni gumb za povratek na prejšnjo stran v brskalniku, bo zapustil spletno stran.

Tu seveda ne gre za dobesedno preprečevanje, pač pa za opozorilo, kot je razvidno iz slike 45 na levi strani.

Komplement zgornji funkciji je funkcija `odstraniPreprecevanjeZapuscjanjaStrani()`. Izvede se ob kliku na gumb »Nazaj«. Ko smo na začetnem položaju strani, nam namreč ni v interesu, da bi uporabnika motili pri zapušcanju strani.

```
function NastaviPreprecevanjeZapuscjanjaStrani() {
    window.onbeforeunload = function() { return "Želite zapustiti iskalnik?
        Če želite zgolj novo iskanje, kliknite na zeleni gumb Nazaj.";
    };
}
function OdstraniPreprecevanjeZapuscjanjaStrani() {window.onbeforeunload = null;}
```

3.4.2.1.2 Skripta `izpisRezultatovIskanj.php`

Parametri, ki jih je poslal odjemalec (objekt `xmlhttp`) spletnem strežniku, so skripti dosegljivi preko spremenljivke `$_GET`, ki je v bistvu tabela. Zaradi lažjega dela (da nam ni treba paziti na kodiranje navednic), za vse poslane parametre ustvarimo svoje spremenljivke. Primer:

```
$iskalnaFraza = "";
if(isset($_GET['iskalnaFraza'])){//Z isset() preverimo ali parameter obstaja.
    $iskalnaFraza = $_GET['iskalnaFraza'];
}
```

Ko določimo vse spremenljivke, shranimo podatke o iskanju (katere filtre je uporabnik uporabil) v MySQLovo tabelo `Iskalnik`, ki služi kot arhiv vseh iskanj obiskovalcev (glej podpoglavje 3.3.4.2). Najprej se povežemo z bazo,... uberemo že znan način iz podpoglavja 3.3.3:

```
include("../class.StreznikInBaza.php");
$setupDB1 = vrniKonfiguriranoBazo();
```

Nato sestavimo poizvedbo MySQL za vstavljanje podatkov, ki za vhodne vrednosti vzame vrednosti naših spremenljivk, pridobljenih iz tabele `$GET`. Zatem pošteno poizvedbo preko funkcije `mysql_query()`.

```
$sqlIskalnik=" INSERT INTO
Iskalnik (iskalna_fraza, blagovna_znamka,
         regija, postni_okraj, status_kmetije, ponudba_kmetije,posebnosti)
VALUES   ('$iskalnaFraza', '$blagovnaZnamka',
         '$regija', '$okraj', '$status', '$zivilo', '$posebnosti')";
mysql_query($sqlIskalnik);
print mysql_error (); //Če želimo, lahko izpišemo tudi morebitno napako.
```

Zatem se sestavi druga poizvedba, namenjena izpisu rezultatov iskanj. Katere tabele bomo naslavljali? Podatki, ki smo jih pripravljeni nuditi uporabniku preko iskalnika, so v več tabelah:

Filtri z opcijami	Tabela/e v bazi, ki hrani/jo iskane podatke	Stolpec/stolpci v tabeli v bazi
Iskalna fraza	<code>Ponudnik</code> <code>Kraj</code> <code>Logotip</code>	.. <code>naslov</code> , <code>naziv_ponudnika</code> <code>postna_stevilka</code> , <code>telefon</code> <code>kontaktna_oseba</code> , <code>gsm</code> , <code>e_mail</code> , <code>zdruzenje</code> , <code>eko_ponudba</code> .. <code>ime_kraja</code> .. <code>ime_logotipa</code> , <code>organizacija</code>
Blagovna znamka	<code>Logotip</code>	<code>id_logotipa</code>
Regija	<code>Regija</code>	<code>id_regije</code>
Poštni okraj	<code>Postni_okraj</code> ali <code>Kraj</code>	<code>id_okraja</code>
Status kmetije	<code>Ponudnik</code>	<code>Status</code>
Ponudba kmetije	<code>Ponudnik</code>	<code>eko_ponudba</code>
Posebnosti	<code>Ponudnik</code>	<code>obisk_solarjev</code>

Iz zgornje tabele se še ne da celovito napovedati, katere tabele se pri poizvedbah iskanja naslavljajo vedno, saj moramo vedeti tudi to, katere podatke vedno izpisujemo. Tabela rezultatov ima namreč fiksen format. Primer njenega izgleda na zaslonu prikazuje slika 46..

Slika 46: Primer izpisa rezultatov iskanj na spletni strani iskalnika



Razdelajmo, do katerih tabel v bazi je treba dostopati za izpis stolpcev tabele na spletni strani:

Stolpec tabele rezultatov na strani	Tabela/e v bazi , ki hrani/jo iskane podatke	Stolpec/stolpci v tabeli v bazi
Regija, polni naslov	Regija Ponudnik Kraj	ime_regije naslov, postna_stevilka ime_kraja
Kontaktna oseba	Ponudnik	kontaktna_oseba,
Naziv in podrobnosti	Ponudnik	naziv_ponudnika, permalink
Kontakt	Ponudnik	telefon, gsm, e_mail, spletna_stran
Združenje	Ponudnik	zdruzenje
Čas in kraj prodaje	Ponudnik	kraj_cas_prodaje
Pridelki, živila	Ponudnik	eko_ponudba

Iz tabele lahko razvidimo, da se neglede na to, kakšne iskalne filtre obiskovalec določi, pri poizvedbi za iskanje vedno naslavlja tabele Ponudnik, PredstavitevPonudnika, Regija, Kraj. Če vse skupaj analiziramo, lahko razberemo, da se v primeru, ko je izbran filter iskalna fraza, naslavlja še tabelo Logotip. Prav tako se dodatno naslavlja tabelo Logotip, če je izbran filter »Blagovna znamka«. To upoštevamo pri sestavljanju poizvedbe SQL.

```
$poizvedbaSql="
SELECT p.*, k.ime_kraja, r.id_regije, r.ime_regije,pr.permalink
FROM Ponudnik p";
$poizvedbaSql.="
LEFT JOIN PredstavitevPonudnika pr ON p.id_ponudnika = pr.id_ponudnika
LEFT JOIN Ponudnik_znamka pz ON p.id_ponudnika = pz.id_ponudnika
LEFT JOIN Kraj k ON p.postna_stevilka = k.postna_stevilka
LEFT JOIN Regija r ON r.id_regije = k.id_regije_predvid";
if ( $iskalnaFraza != "" or $blagovnaZnamka != "" ){ $poizvedbaSql.="
LEFT JOIN Logotip l ON l.id_logotipa = pz.id_logotipa";
}
$poizvedbaSql.= " WHERE p.vidnost='1' AND p.certifikat='1' ";
if ($regija != 0){ $query= $query . " and (r.id_regije = '". $regija. "' )"; }
if ($okraj != 0){ $query= $query . " and (k.id_okraja = '". $okraj. "' )"; }
if ($status != '0'){ $query= $query . " and (p.status = '". $status. "' )"; }
if ($zivilo != '0'){ $query= $query . " and (p.eko_ponudba like '%" $zivilo. "%' )"; }
if ($blagovnaZnamka != '0'){ $query= $query . " and (l.id_logotipa='". $blagovnaZnamka. "' )"; }
if ($posebnosti == '1'){ $query= $query . " and (p.obisk_solarjev = '1' )"; }
```

V naslednjem delu moramo določiti pogoje WHERE poizvedbe za iskalno frazo. Upoštevamo:

- Za izpis kmetij, ki ponujajo hkrati jabolka in hruške, vnesemo »jabolka & hruške«.
- Za izpis kmetij, ki ponujajo bodisi jabolka bodisi hruške, vnesemo »jabolka,hruške«.
- Če bomo vpisali niz »jabolka hruške«, bo iskalnik vmes vstavil vejico.
- Operator IN ima prednost pred ALI. Če vpišemo niz »jabolka & sir, med«, bo najprej niz najprej razdeljen na dva dela, pri čemer bo ločilo vejica. Prvi del: »jabolka & sir«. Drugi del: »med«. Oba dela bosta posamično ovita v oklepaj in zaklepaj, vmes pa bo zapisan

operator ALI. Niz »jabolka & sir« bo obravnavan posebej. Znak & bo nadomeščen z operatorjem IN. Kot rezultat dobimo: ((jabolka IN sir) ALI med).

```
if ($iskalnaFraza != ""){
    //V Iskalno frazo vpeljemo operatorja IN (znak $) ter ALI (vejica)
    $iskalnaFraza = urldecode($iskalnaFraza); //Ker je bil s strani JavaScript zakodiran s
    //funkcijo encodeURIComponent()... ampak lahko funkcionira tudi brez tega.
    //Najprej razdelimo iskalno frazo na več delov, glede na vejico.
    $besedeALI = preg_split("/[,]+/", $iskalnaFraza, -1, PREG_SPLIT_DELIM_CAPTURE |
PREG_SPLIT_NO_EMPTY);
    $poizvedbaSql = $poizvedbaSql . " AND ("; // Ker imamo še druge filtre
    $prvaBesedaOR = true; //Prva beseda je izjema in se jo obravnava drugače.
    foreach ($besedeALI as $besedaALI) {
        $besedaALI=trim($besedaALI); // Odstranimo odvečne presledke.
        $besedaALI=str_replace(" ", "", $besedaALI); //Vse presledke nadomestimo z vejicami
        $besedaALI = preg_replace('/',+',', $besedaALI); //Znebimo se podvojenih vejic.
        $besedaALI = preg_replace('&+', '&', $besedaALI); //Znebimo se podvojenih &.
        $besedaALI = str_replace("&", "&", $besedaALI); //Odpravimo vsiljeno napako.
        if ($besedaALI=="") { continue; } //Beseda po obdelavi ne sme biti prazna.
        if (! $prvaBesedaOR) { //Vejica v iskalni frazi predstavlja operator OR.
            $poizvedbaSql = $poizvedbaSql . " OR "; // operator ALI v obliki mySQL ukaza
        }
        $prvaBesedaOR = false;
        $besedeIN = preg_split("/[&]+/", $besedaALI, -1, PREG_SPLIT_DELIM_CAPTURE |
PREG_SPLIT_NO_EMPTY); //Dele med vejicami razdelimo na več delov glede na znak &.
        $prvaBesedaIN = true;
        $poizvedbaSql .= "(";
        //echo("<BR/>"); // Če bi želeli videti, kako deluje algoritem.
        foreach ($besedeIN as $besedaIN) {
            $besedaIN=trim($besedaIN); //Odstranimo odvečne presledke.
            $besedaIN=trim($besedaIN, ','); //Odstranimo še odvečne vejice.
            if (! $prvaBesedaIN) {
                $poizvedbaSql = $poizvedbaSql . " AND "; //Operator IN v obliki mySQL ukaza.
                //echo("IN"); // Če bi želeli videti, kako deluje algoritem.
            }
            //echo($besedaIN); // Če bi želeli videti, kako deluje algoritem.
            $prvaBesedaIN = false;
            $poizvedbaSql = $poizvedbaSql . " (
                p.naslov LIKE '%" . $besedaIN . "%'
                OR p.postna_stevilka LIKE '%" . $besedaIN . "%'
                OR p.kontaktna_oseba LIKE '%" . $besedaIN . "%'
                OR p.naziv_ponudnika LIKE '%" . $besedaIN . "%'
                OR p.telefon LIKE '%" . $besedaIN . "%'
                OR p.gsm LIKE '%" . $besedaIN . "%'
                OR p.e_mail LIKE '%" . $besedaIN . "%'
                OR p.združenje LIKE '%" . $besedaIN . "%'
                OR p.eko_ponudba LIKE '%" . $besedaIN . "%'
                OR l.ime_logotipa LIKE '%" . $besedaIN . "%' )" ;
                OR l.organizacija LIKE '%" . $besedaIN . "%' )" ;
            }
            $poizvedbaSql .= ")";
        } $poizvedbaSql = $poizvedbaSql . " )"; }
}
```

Sedaj imamo poizvedbo v spremenljivki `$poizvedbaSql`. Postredujemo jo strežniku MySQL v obdelavo. Njegov odgovor je spisek vrstic, katere formatirano izpisujemo kot tabelo HTML, narejeno z elementi `<div>`. Oblikovnega dela tu podrobno ne obravnavamo, koncept izpisa pa je takšen:

```
$rowIndex = 0;
if ($ponudniki = mysql_query($poizvedbaSql)) {
    $vrsticaLihaSoda = "soda";
    while($row = mysql_fetch_assoc($ponudniki)) {
        if ($rowIndex % 2) {
            $vrsticaLihaSoda="soda";
        } else{
            $vrsticaLihaSoda="liha";
        }
        $rowIndex = $rowIndex +1;
        $izpis.=... //Sledi spuščeni del - formatiran izpis tabele.
        //Do elementov vrstice dostopamo na tak način: $row["telefon"]
    }else{ $izpis.="<p>".mysql_error(). "</p>"; } //Izpis morebitne napake.
    //Naš odgovor odjemalcu je sestavljen iz dveh delov.
    //V prvem je tabela z rezultati iskanj, v drugem pa je št. zadetkov iskanj.
    //Da bo lahko funkcija stateChangedPonudniki() (glej naslednje podpoglavje)
    // razbrala oba dela, vpeljemo ločilni niz, na podlagi katerega bo odgovor
    // razdelila na dva dela.
    $izpis.="|$|";
    $izpis.="Število zadetkov: ".$rowIndex;
    echo $izpis; //Na tak način skripta vrne odgovor odjemalcu.
    mysql_close($setupDB1->vрниPovezavo()); //Zapremo povezavo s strežnikom mySQL.
```

3.4.2.1.3 Odjemalec preko funkcije `stateChangedPonudniki()` obdela odgovor

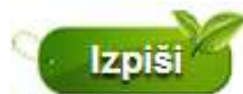
Zatem, ko spletni strežnik do konca sprocesa datoteko `izpisRezultatovIskanj.php`, pošlje odjemalcu (objektu `xmlhttp`) odgovor, oziroma rezultate iskanj. Ta rezultat je navadno besedilo (navaden niz, sestavljen iz znakov). Ker smo v besedilu poslali dva različna odgovora, ki jih želimo prikazati na različnih mestih, moramo niz razdeliti na dva dela. To znotraj funkcije `stateChangedPonudniki()` naredimo s funkcijo `split`, ki poišče zaporedje znakov `|$|`, ki smo jih umestili v odgovor namerno.

```
//Funkcija, ki se izvede, ko objekt xmlhttp dobi odgovor od spletnega strežnika.
function stateChangedPonudniki() {
    if (xmlhttp.readyState==4) {
        var update = new Array();
        if(xmlhttp.responseText.indexOf('|$|' != -1)) {
            update = xmlhttp.responseText.split('|$|');
            // Array bo vseboval update[0] = rezultati iskanj,
            //                               update[1] = število rezultatov iskanj.
            document.getElementById("ResponseHolder").innerHTML=update[0];
            document.getElementById("stevilo-zadetkov-iskanj").innerHTML=update[1];
        }else{ }
    }
}
```



3.4.3 Prikazani podatki v rezultatih iskanj

Povzemimo vse skupaj. Ko je spletna stran že naložena, lahko uporabnik preko klika na gumb »Izpiši« (glej sliko 47) dobi na zaslonu izpisane rezultate iskanj (glej sliko 48) na isti strani, s tem da v ozadju delujejo funkcije skrivanja elementov forme, katerih ne bomo obravnavali.

Slika 47: Gumb za izpis rezultatov



Slika 48: Primer izpisa rezultatov iskanj

Stolpce, ob katerih je ikona , se lahko razvrsti po abecednem vrstnem redu. Ob kliku na specifično ikono se vsi rezultati razvrstijo na račun funkcije iz knjižnice jQuery. Zgled:

```
$( '.tableRow' ).sort( function ( a, b ) {
    if ( $( columnId ).hasClass( "sortAtoZ" ) ) {
        x = $(a).find(columnClass).data('tosort').localeCompare (
            $(b).find(columnClass).data('tosort') );
    } return x; }
).each(function ( _, tableRow ) { $(tableRow).parent().append(tableRow); });
```

Podrobnosti polj v stolpcih:

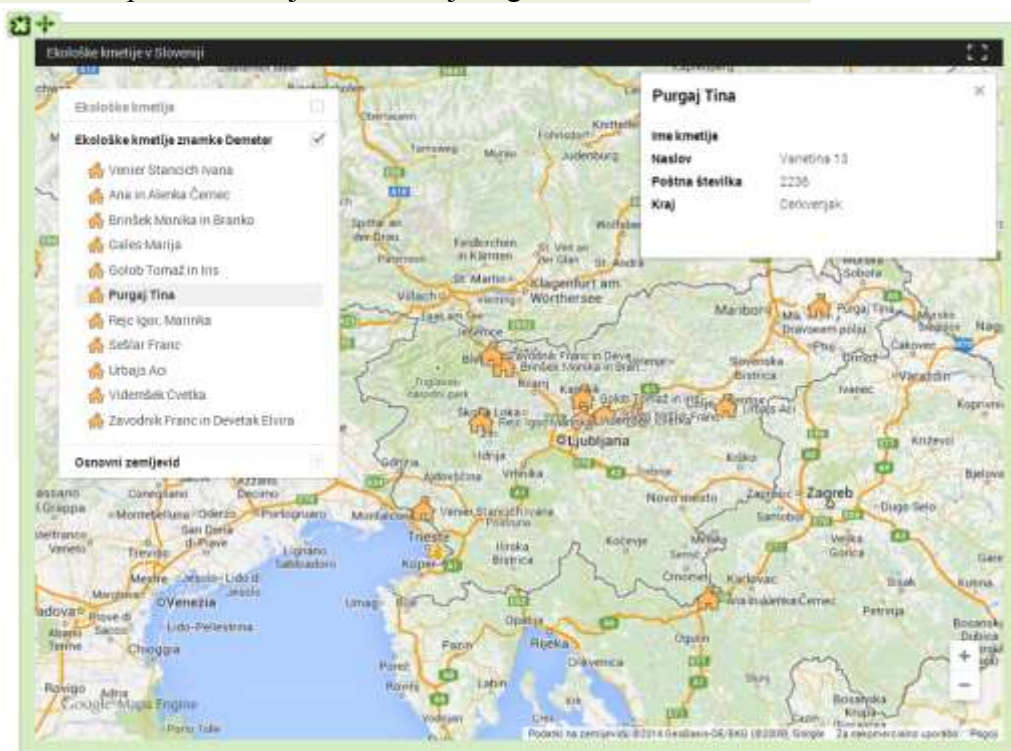
Polje »regija, polni naslov« je hkrati povezava na lokacijo tega naslova na storitev Google Maps.	Polje »naziv in podrobnosti« vsebuje več povezav: <ul style="list-style-type: none"> Podrobnosti – predstavitevna stran kmetije na našem spletišču. Povezava do certifikata na stran Ministrstva za kmetijstvo in okolje. Lastni spletni naslov kmetije, če obstaja. 	Polje »pridelki, živila« je pojavni menu (pojavi se, ko se kurzor miške pojavi nad poljem).
---	---	---

3.4.4 Zemljevid ekoloških kmetij

Zemljevid ekoloških kmetij je izdelan z aplikacijo Google Maps Engine Lite. Uporaba te aplikacije je dovoljena za ustvarjanje zemljevidov za nekomercialno uporabo. To pomeni, da za uporabo aplikacije ne smemo prejeti plačil ali nadomestil od kogar koli v kakršni koli obliki. V različici Lite smo omejeni s številom nivojev, atributov, vrstic v dokumentu CSV (Comma Separate Value), ki ga uvažamo, itd. Različica Lite nam omogoča, da lahko na zemljevidu kmetije kategoriziramo v 3 kategorije (3 nivoje na zemljevidu). Za naše potrebe je to trenutno ravno prav, saj imamo 3 blagovne znamke: Demeter, Biodar, Ekodar. Različica Pro, katere najemnina stane 50 dolarjev letno, omogoča več nivojev.

Na zemljevidu (glej sliko 49) lahko uporabnik filtrira prikaz ekoloških kmetij glede na certifikat.

Slika 49: Primer prikaza zemljevida kmetij blagovne znamke Demeter



Ikone na zemljevidu predstavljajo ekološke kmetije, ki posedujejo specifičen certifikat. Po kliku na ikono na zemljevidu se v oblaku prikažejo osnovni podatki o kmetiji.

Za posodobitev podatkov so na Googlu ponudili enostavno rešitev – uvoz podatkov iz datoteke CSV. Nastaviti moramo le, kateri od stolpcev predstavljajo attribute, ki se nanašajo na lokacijo. Poleg teh lahko vpeljemo še dodatne attribute, ki se lahko izpišejo v oblaku nad izbrano lokacijo (kmetijo). To je lahko na primer povezava do spletne strani ekološke kmetije. Datoteko s podatki izvozimo iz podatkovne baze spletnega iskalnika ekoloških kmetij.

Zemljevid se ne naloži hitro, zato je urejeno, da se ob prihodu na stran ne naloži. Naloži se le koda njegovega ogrodja. Vsebina pa se naloži po za tem, ko uporabnik klikne na ikono zemljevida. Ogrodje zemljevida je v osnovi nekaj gnezdenih elementov `<div>`:


```

<div class="ekoZemljevid"> <!-- Ikona zemljevida -->
  <div onclick="javascript:showDialogEcoSite('#zemljevid');">
    
  </div></div>
<div id="zemljevidDrag"> <!-- Ogrodje zemljevida -->
  <div id="zemljevid" class="dialogWindow" title="Basic dialog"
    style="display:none">
    <div class="buttons"> <div class="iconClose"
      onclick="javascript:closeDialogEcoSite('#zemljevid');"></div>
    <div class="iconDrag" id="zemljevidIconDrag"></div>
    </div>
    <div class="dialogBodyMap"> <div id="zemljevidFrame">
      <!-- Sem se preko poizvedbe Ajax naloži zemljevid. -->
      <div id="mapFrameDiv" ></div>
    </div></div>
  </div></div>

```

Vendar to ni njegova končna oblika kode, saj se mu na račun uporabe funkcije dialog() iz knjižnice jQuery ob kliku na element `#zemljevid` dodajo še dodatni deli. Na koncu dobimo okno, katerega lahko premikamo po zaslonu ali zapremo.

```

function showDialogEcoSite(id){$(id).dialog( "open" ); $(id).show();}
function closeDialogEcoSite(id){ $(id).dialog('close');}
$(function(){//Inicijalizacija pojavnega okna se sproži ob prihodu na stran.
  initMapDialog("#zemljevid","#zemljevidDrag","#zemljevidIconDrag");
});
function initMapDialog(id,idDrag,idIcon){
  if ($(id).length){ $(id).draggable({ handle: idIcon });
    $(id).dialog({
      appendTo: idDrag,//Določimo element, preko katerega okno premikamo.
      autoOpen: false, //Določimo, da se okno ne odpre samodejno.
      width: ($(window).width()-15),
      position: { my: "center", at: "top", of: window },
      dialogClass: 'noTitleStuff', //Določimo poseben razred
      open: function( event, ui ) { procesirajZemljevid(); },//Rokovalnik.
      resizable: false, //Določimo, da se oknu ne da spreminjati velikosti.
      cache: true, //Določimo, da se vsebina shrani v predpomnilnik.
      show:{effect:"none", duration:0}, hide:{effect:"none",duration:0}
    });
  }
}

```

Ko imamo okno, potrebujemo še vsebino – zemljevid. Ta se v okno naloži z uporabo poizvedbe Ajax. Najprej

```

function procesirajZemljevid(){//Rokovalnik izvede poizvedbo Ajax.
  xmlhttp=GetXmlHttpRequest();
  if (xmlhttp==null){alert ("Brskalnik ne podpira XMLHttpRequest!"); return; }
  xmlhttp.onreadystatechange=stateChangedZemljevid;
  xmlhttp.open("GET","./zemljevid-eko-kmetij.htm",true);//kličemo dat.
  xmlhttp.send(null);
  return;}
function stateChangedZemljevid(){//Ob ustreznem odgovoru s strežnika
  if (xmlhttp.readyState==4) { //naložimo zemljevid v element mapFrameDiv
    document.getElementById("mapFrameDiv").innerHTML=xmlhttp.responseText;
    jQuery("#mapFrameDiv").css("height",$(window).height() -100);  }}

```

3.4.5 Prijava in registracija



Slika 50: Zavihek za prijavo

V aplikacijo se lahko prijavijo uporabniki, to so pridelovalci ekoloških živil, oziroma osebe, katerim so pridelovalci zaupali uporabniško ime in geslo spletne aplikacije. Blok (gradnik) s prijavno formo se prikaže dinamično, po kliku na zavihek v zgornjem delu okna (glej sliko 50). Za prikaz se uporablja knjižnica jQuery, ki omogoča premikanje bloka z miško, zamegljevanje teksta ob kliku na vnosna polja, posebne grafične efekte ob zaprtju bloka.

Uporabniki imajo svoj uporabniški konto, ki hrani naslednje podatke:

- Id številko pridelovalca živil
- Uporabniško ime
- Uporabniško geslo
- Ime in priimek
- E-mail in telefon
- Validacijska koda (pogojno)

E-mail ne more nadomestiti uporabniškega imena, saj se prvi lahko spreminja, med tem ko uporabniško ime ostaja.

Vsako leto za tem, ko Ministrstvo za kmetijstvo objavi nov seznam ekoloških pridelovalcev, pošljemo vsem pridelovalcem, za katere smo pridobili e-naslov, elektronsko sporočilo. Sestavni del sporočila so podatki o uporabniškem kontu in povabilo v administrativni del spletne aplikacije, kjer lahko ustvarijo svojo predstavitevno stran.

V primeru, da izgubijo bodisi uporabniško ime bodisi geslo, lahko v prijavni formi (glej sliko 51) vpišejo e-mail, na katerega se pošlje sporočilo z uporabniškim imenom in geslom, povezanim z vpisanim e-naslovom.

Slika 51: Forma za prijavo v uporabniški račun

The image shows a web form titled "Login forma (za zastopnike eko kmetij)". It has a light green background with a subtle pattern of leaves. At the top left, there is a small icon of a leaf and a plus sign. On the right side, there is a vertical green button labeled "Register" with a leaf icon. The form contains two input fields: "Uporabniško ime:" with a person icon and "Geslo:" with a padlock icon. Below these fields is a link that says "Ne veste gesla/uporabniškega imena? Vpišite e-mail in vam bodo podatki poslani." with an envelope icon. At the bottom, there is a large black button labeled "Potrdi" with a leaf icon.

Problem je, ko (če) s strani Ministrstva za kmetijstvo in gozdove ne pridobimo e-naslovov za vse pridelovalce. Poleg tega ne izločamo možnosti, da nekateri niso več v uporabi. To je problematično s tega vidika, da takšnim uporabnikom ne moremo poslati uporabniškega imena in gesla preko elektronske pošte. Kljub temu pa jim želimo omogočiti, da lahko spreminjajo podatke v svojem kontu.

Problem prejšnjega odstavka rešuje registracijska forma, ki jo prikazuje slika 52.

Slika 52: Forma za registracijo

Omogoča, da ponudnik vpiše svoje kontaktne podatke, določi svoje uporabniško ime in geslo, ter sproži postopek registracije. E-naslov osebe, ki se je prijavila, moramo nekako povezati z obstoječim pridelovalcem živil v bazi. Oseba tega ne sme imeti možnost storiti sama, saj bi lahko prišlo do pretvarjanja s strani tretje osebe.

3.4.5.1 CAPTCHA

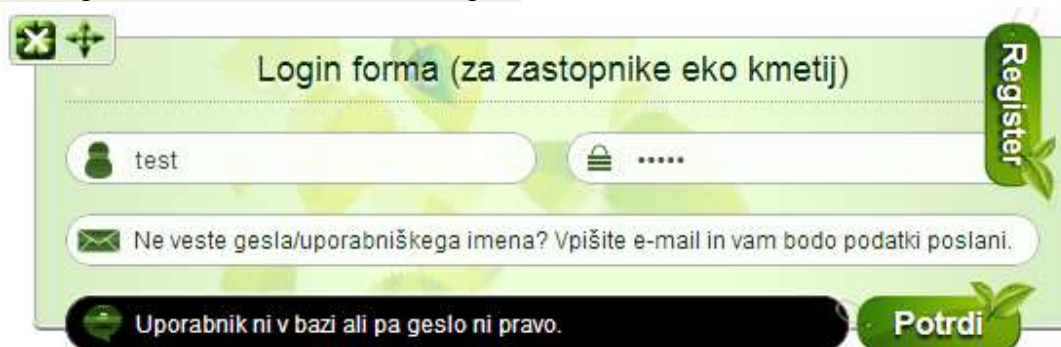
Za preprečitev email spammanja s strani računalniških programov je registracijska forma zaščitena s CAPTCHO, opisano v podpoglavju 3.4.7. Uporabimo jo za zaščito pred tem, da bi nam kakšen program množično ustvarjal uporabniške konte in zasičil podatkovno bazo. Konto je namreč ustvarjen takoj po uspešni potrditvi registracijske forme, le aktiviran ni. S tem se izognemo tudi nezaželenemu obremenjevanju poštnega strežnika.

3.4.5.2 Poizvedbe mySQL z uporabo tehnologije Ajax

V registracijski formi se pri vnosu uporabniškega imena z uporabo AJAXa (kliče se php datoteko) sproti preverja, ali uporabniško ime že obstaja v bazi; izpišejo se podobna že obstoječa uporabniška imena. V formi login se obstoj up. imena preverja po kliku na polje za vnos gesla). Podatki se črpajo iz baze MySQL. Tabela `Uporabnik` je indeksirana za potrebe hitrega iskanja.

Ob kliku na gumb "Potrdi" vmesnik sproti obvešča uporabnika o ustreznosti vnosov, napakah, dogodkih. Obvestila se prikažejo v črno označenem območju. Primer takšnega izpisa prikazuje slika 53.

Slika 53: Izpis obvestila o neustreznosti vpisa



3.4.5.3 Uspešna registracija – potrditveni mail

Po uspešni registraciji se na e-naslov, povezan z ustvarjenim kontom, pošlje informativno sporočilo s postopkom za aktivacijo uporabniškega konta:

Vaš uporabniški konto bo aktiviran, ko se prepričamo, da ste resnično kontaktna oseba oziroma zastopnik kmetije, za katerega se izdajate. S tem želimo preprečiti možnost zlorab pretvarjanja s strani tretje osebe. Ker vašega e-maila še nimamo v bazi, lahko za identifikacijo uporabite eno od treh možnosti:

- *Prosimo vas, da s telefonske številke <telefon> pošljete SMS z vsebino PRIJAVA. V primeru ujemanja telefonske številke bo validacija uspešna in boste prejeli e-sporočilo o uspešni aktivaciji vašega konta.*
- *V primeru, da s te številke ne morete poslati SMS-a, nas lahko pokličete v času uradnih ur <uradneUre>..*
- *V primeru, da telefonske številke <telefon> več ne uporabljate, nam lahko odgovorite na to sporočilo s prošnjo, da vam po fizični pošti pošljemo validacijsko kodo. Slednjo nam boste posredovali nazaj bodisi preko elektronske pošte bodisi preko SMS-a. Ko bo administrator kodo preveril, boste o uspešni aktivaciji vašega konta obveščeni preko e-maila.*

Na mestih <telefon> in <uradneUre> skripta PHP `posljiEMailRegistracijo.php`, ki služi za sestavljanje in pošiljanje sporočil, vstavi uporabniške in administratorjeve podatke iz tabel `Uporabnik` in `Admin`. Če se uporabnik odloči za tretjo možnost, administrator v polje `validacijska_koda` iz tabele `Uporabnik` vpiše poljubno kodo.

3.4.5.4 Uspešen vpis – prehod na uporabniške administrativne strani

Ob uspešnem vpisu se blok prijave forme samodejno zapre. V brskalniku se v novem zavihku odpre administrativna stran uporabniškega konta, kjer lahko uporabnik:

- Spremeni kontaktne podatke (telefon, e-mail, naslov spletne strani)
- Spremeni podatke o času in kraju prodaje
- Izdela oblikovano opisno besedilo o kmetiji, ponudbi,..., formatirano v HTML

3.4.6 Predstavitvene strani in administrativna stran za uporabnike

Predstavitvene in administrativne strani so si precej podobne, s tem da na administrativnih za razliko od predstavitvenih ni zemljevida, na predstavitvenih pa za razliko od administrativnih ni možno spreminjati in shranjevati vsebine nekaterih polj.

3.4.6.1 Predstavitvene strani

Zgled predstavitvene strani kmetije prikazuje slika 54.

Slika 54: Predstavitvena stran kmetije

The screenshot shows the website 'Ekološke kmetije po Sloveniji' with a 'Prijava' button in the top right. The main content area displays the profile of 'Ekološka kmetija Ronkaldo'. It includes contact information (Kontaktna oseba: Adamič Miran, Telefon, E-mail: info@ronkaldo.si, Naslov: Beblerjeva ulica 10, 6000 Koper/Capodistria) and product details (Naziv ponudnika: Kmetija Ronkaldo, Gam: 041-624-639, Spletna stran: http://www.ronkaldo.si, Certificat: informacije o certifikatu). A section titled 'Opis kmetije' describes the product as 'EKSTRA DEVIŠKO OLJČNO – OLIVNO OLJE »RONKALDO« Z ZAŠČITENO OZNAČBO POREKLA (Z.O.P.) SLOVENSKE ISTRE'. Below this is a map showing the location at 'Beblerjeva ulica 10, 6000 Koper/Capodistria'. The footer contains logos for various organic certification bodies (BIO, EKO, demeter, etc.) and the author's name, Boris Turk.

V podpoglavju 3.3.4.10 je opisana podatkovna tabela `Predstavitev_ponudnika`, v kateri so shranjeni relativni spletni naslovi (v stolpcu `permalink`) in vsebina predstavitvenih strani (stolpec `predstavitev`) za vsako kmetijo posebej. Permalink je sicer splošen izraz za permanentno povezavo. V praksi ima lahko vsaka stran eno permanentno povezavo in več drugih, ki se preko preusmeritvenih mehanizmov strežnika preusmerijo na permanentno.

Predstavitvene strani uporabnikov so dinamične strani in v praksi ne obstajajo kot statični dokumenti. Njihova vsebina obstaja le v podatkovni tabeli `Predstavitev_ponudnika`. Spletne strani pa se ustvarijo na zahtevo – ob vsakem dostopu do naslova, ki je dodeljen kmetiji. Potek opisujemo na primeru testne kmetije, ki se navidezno nahaja v imeniku `./pridelovalci/test`

V osnovi ob poskusu dostopa na spetni naslov `http://ekološke-kmetije.si/pridelovalci/test` spletni strežnik vrne napako 404 – stran ne obstaja, saj v imeniku `./pridelovalci` ne obstaja noben dokument z imenom `test`. Mi pa želimo, da ob takšnem dostopu spletni strežnik vrne vsebino kmetije `test`.

To omogočata dve datoteki:

- `predstavitevKmetije.php`
- `.htaccess`

Datoteki se nahajta v imeniku `./pridelovalci`. Na račun datoteke `.htaccess` lahko dostop do imenika `./pridelovalci/test` preusmerimo na datoteko `predstavitevKmetije.php`, v kateri so navodila o tem, za katero kmetijo naj se izpiše vsebina v standardizirani obliki predstavitvene spletne strani. To naredimo tako, da napišemo preusmeritveno pravilo – `RewriteRule`. Da nam ne bi bilo potrebno takšnih pravil pisati za vsako kmetijo posebej, smo v datoteki `.htaccess` pri preusmeritvenem pravilu uporabili regularni izraz, ki vse dostope do navideznih datotek v imeniku `./pridelovalci` preusmeri na naslov [http://ekološke-kmetije.si/pridelovalci/predstavitevKmetije.php?kmetija=\\$1](http://ekološke-kmetije.si/pridelovalci/predstavitevKmetije.php?kmetija=$1), pri čemer je `$1` niz, ki sledi imeniku predstavitev. V našem primeru je to niz »test«.

Vsebina datoteke `.htaccess` je takšna:

```
Options +FollowSymlinks
RewriteEngine on
RewriteRule ^([a-zA-Z0-9\-]+)$ http://ekološke-
kmetije.si/pridelovalci/predstavitevKmetije.php?kmetija=$1 [L,NC]
```

Z uporabo pravil v datoteki `.htaccess` torej dosežemo, da spletni strežnik ob tem, ko dobi zahtevo za dostop do strani

`http://ekološke-kmetije.si/pridelovalci/test`, zahtevo preusmeri na

`http://ekološke-kmetije.si/pridelovalci/predstavitevKmetije.php?kmetija=test`.

Nato začne interpretirati vsebino datoteke `predstavitevKmetije.php`, kateri posreduje parameter `kmetija` z vrednostjo »test«.

Naloga skripte `predstavitevKmetije.php` je, da sestavi dokument HTML s predstavitveno vsebino specifične kmetije. Stvar je podobna kot pri datoteki `index.php` (glej podpoglavje 3.3.3).

Najprej iz spremenljivke `$_GET` ugotovi, katero kmetijo mora obdelati, zatem pa na osnovi tega sestavi poizvedbo SQL, ki vrne podatke o kmetiji. Poizvedba je na naslednji strani.

```

$kmjetija ="Testna";
if(isset($_GET['kmetija'])){ //Parameter kmetija je podan v .htaccess.
    $kmetija = $_GET['kmetija']; }//Konec if.
$query="SELECT  p.id_ponudnika, p.kontakt_oseba, p.naziv_ponudnika,
p.telefon, p.gsm, p.e_mail, p.spletna_stran, p.naslov, concat
(p.postna_stevilka, ' ', k.ime_kraja) as posta , lower(p.eko_ponudba)
eko_ponudba, p.kraj_cas_prodaje, p.certifikat, pr.permalink,
pr.predstavitev, pr.staraPredstavitev,p.vidnost,p.obisk_solarjev
FROM  PredstavitevPonudnika  pr
LEFT JOIN Ponudnik p on p.id_ponudnika = pr.id_ponudnika
LEFT JOIN Ponudnik_znamka pz on p.id_ponudnika = pz.id_ponudnika
LEFT JOIN Logotip l on l.id_logotipa = pz.id_logotipa
LEFT JOIN Kraj k on p.postna_stevilka = k.postna_stevilka
LEFT JOIN Regija r on r.id_regije = k.id_regije
WHERE pr.permalink = ' ".$kmetija ."'";
$naslovKmetije = ""; $row = "";
if ($rezultat = mysql_query($query)) {
    while($tempRow = mysql_fetch_assoc($rezultat)) {
        if ($tempRow["naziv_ponudnika"] <> '') {
            $naslovKmetije = str_replace(array("Kmetija","kmetija",
            "KMETIJA"), "", $tempRow["naziv_ponudnika"]);
        } else { $naslovKmetije = $tempRow["kontakt_oseba"]; };
        $predstavitev = $tempRow["predstavitev"];
        $row = $tempRow;
    } //konec while
} else { echo ("<p>".mysql_error (). "</p>"); } //Izpis morebitne napake

```

Nato se na podoben način, kot je bilo opisano za datoteko `index.php` v podpoglavju 3.3.3, izpiše glava dokumenta HTML, nato formatirana vsebina na osnovi podatkov, ki jih dobimo iz zgornje poizvedbe, potem pa še noga strani – logotipi.

Pri tem omenimo, da je vsebina oznak meta prilagojena za vsako kmetijo. V `<title>` je zapisan naziv kmetije, v `<meta description...>` je zapisana kontaktna oseba, naslov kmetije in ponudba, v `<meta keywords...>` so našeta živila v ponudbi.

Del predstavitev strani sta zemljevid in povezava do certifikata.

3.4.6.1.1 Kako dobimo lokacijo kmetije na zemljevidu Google Maps?

Na osnovi znanih koordinat vdelamo zemljevid na stran z uporabo kode iz [20]:

```

<script type="text/javascript">var geocoder;var map;var a;
function splosna_inicializacija_zemljevida() {
    var myOptions = {zoom:14,center:new google.maps.LatLng(40.805478,-
73.96522499999998),mapTypeId: google.maps.MapTypeId.ROADMAP};
    map = new google.maps.Map(document.getElementById("gmap_canvas"),
myOptions);
    marker = new google.maps.Marker({map: map,position: new
google.maps.LatLng(40.805478, -73.96522499999998)});
    infowindow = new google.maps.InfoWindow({content:"<b>The
Breslin</b><br>2880 Broadway<br> New York" });
    google.maps.event.addListener(marker, "click",
function() {infowindow.open(map,marker);});infowindow.open(map,marker);}
</script>

```

Rešitev s prejšnje strani za nas ni ustrezna, saj točnih koordinat kmetij ne poznamo. Spodnja funkcija razširja funkcijo `splosna_inicializacija_zemljevida()` na način, da nam ni potrebno določiti koordinat, ampak uporabimo objekt `geocoder`, čigar funkcija `geocode()` nam vrne koordinate iskanega naslova.

```
<script type="text/javascript">
function inicializacija_zemljevida() {
    var myOptions = { zoom: 10, center: latlng, mapTypeControl: true,
    mapTypeControlOptions: {style:
        google.maps.MapTypeControlStyle.DROPDOWN_MENU}, navigationControl:
        true, mapTypeId: google.maps.MapTypeId.ROADMAP
    };
    map = new google.maps.Map(document.getElementById("gmap_canvas"),
    myOptions);
    //Marker, infowindow in google.maps.event bomo urejali znotraj geocoder-ja,
    // ki nam bo vrnil koordinate glede na specifičen naslov.
    geocoder = new google.maps.Geocoder();
    var latlng = new google.maps.LatLng(-34.397, 10.644); //to se spremeni
    if (geocoder) { geocoder.geocode( { 'address': address},
    function(results, status) {
        if (status == google.maps.GeocoderStatus.OK) {
            if (status != google.maps.GeocoderStatus.ZERO_RESULTS) {
                map.setCenter(results[0].geometry.location);
                var marker = new google.maps.Marker({position:
                results[0].geometry.location, map: map, title: address });
                var myContent = '
                <div style="text-align:center; width:350px;
                height:50px; "><b>'+address+'</b><br>
                Ekološka kmetija <?php echo $naslovKmetije; ?> </div>';
                var infowindow = new google.maps.InfoWindow({ content: myContent});
                google.maps.event.addListener(marker, 'click', function() {
                    infowindow.open(map,marker);}); infowindow.open(map,marker);}
                else {alert("Ni podatkov");}
            } else {alert("Geocode zemljevida ni mogoče prikazati. " + status);
            }
        }
    });
    //Kličemo funkcijo, s katero nastavimo poslušalca DOM, ki bo za tem, ko se
    // bo stran naložila, avtomatično izpisal zemljevid.
    google.maps.event.addDomListener(window, 'load', inicializacija_zemljevida);
</script>
```

3.4.6.1.2 Kako ustvarimo povezavo do certifikata pridelovalca živil?

Zgled povezave, ki jo lahko generira koda na naslednji strani iz naziva ponudnika ali primka kontaktne osebe: <https://webapl.mkgp.gov.si/bioWeb/isci.php?isci=Adamič>. Ko obiskovalec strani klikne na takšno povezavo, je preusmerjen na stran iskalnika Ministrstva za kmetijstvo in okolje (glej sliko 55).

Slika 55: Primer izpisa rezultatov iskanja na strani Ministrstva za kmetijstvo in okolje



The screenshot shows the website of the Ministry of Agriculture and Forestry of the Republic of Slovenia. The header includes the ministry's name and the European Union flag. The main content area displays search results for organic farming certificates. A search bar at the top left contains the name 'Adamič'. Below the search bar, there is a table with the following columns: 'Št. certifikata' (Certificate Number), 'Naziv' (Name), 'Pošta' (Post), 'Datum izdaje' (Issue Date), 'Datum veljavnosti' (Validity Date), and 'Poglej' (View). The table contains three rows of data.

Št. certifikata	Naziv	Pošta	Datum izdaje	Datum veljavnosti	Poglej
P2713	Adamič Janez	Črna na Koroškem	27.09.2013	31.12.2014	Poglej
BV-SVN-EKO-138	ADAMIČ VOJKO	ZALOG	15.10.2013	24.10.2014	Poglej
15838	ADAMIČ Miran	Izola - Isola D'Istria	01.08.2013	31.12.2014	Poglej

V podpoglavju 2.1.1.3 smo že pojasnili, kako deluje iskalnik certifikatov. Omejen je na iskanje po priimku in nazivu kmetije. Ker ne hranimo podatkov o številki certifikatov kmetij, je naša povezava na certifikat pravzaprav povezava na iskalnik certifikatov Ministrstva za kmetijstvo in okolje. V povezavi mu posredujemo parameter, ki je bodisi priimek bodisi naziv.

```
<?php $certifikat ="https://webapl.mkgp.gov.si/bioWeb/isci.php?isci=";
if ($row["kontaktna_oseba"] <> ''){ //iščemo ali po priimku
    $delcki = explode(" ", $row["kontaktna_oseba"]); //razdelimo na 2 dela
    $certifikat.=$delcki[0]; //Prvi del je priimek, drugi pa ime
} else { //ali pa po nazivu ponudnika
    $certifikat.=$row["naziv_ponudnika"];
    $search = str_split(' ');
    //presledke v povezavi je treba pretvorit v znak +
    $certifikat= str_replace($search, "+", $certifikat);}?>
```

3.4.6.2 Administrativne strani certificiranih uporabnikov – delo s sejami

Pri prijavljanju v administracijo se preverjanje uporabniškega imena in gesla izvaja asinhrono preko poizvedbe Ajax. Kliče se funkcija `vrniObstojUpImenaInGeslo()`:

```
function vrniObstojUpImenaInGeslo(upIme,geslo) {
    xmlhttpObstojUpImenaInGeslo=GetXmlHttpRequest();
    if (xmlhttpObstojUpImenaInGeslo==null){alert("XMLHTTP ni odprt!");return;}
    var url= baseDir +
        "moduli/PrijavaInRegistracija/uporabnik/aliJeUporabnikInGesloVBazi.php";
    url=url+"?q="+upIme+"&g="+kriptirajGeslo(geslo);
    xmlhttpObstojUpImenaInGeslo.onreadystatechange =
        stateChangedObstojUpImenaInGeslo; //Določimo rokovalnik poizvedbe Ajax
    xmlhttpObstojUpImenaInGeslo.open("GET",url,true);
    xmlhttpObstojUpImenaInGeslo.send(null);}
```

Zgornja funkcija ustvari objekt `XMLHttpRequest` (glej podpoglavje 3.4.2.1.1), ki kliče oddaljeno datoteko `aliJeUporabnikInGesloVBazi.php` z dodatnima parametroma. Datoteko začne spletni strežnik interpretirati. Bistvo te datoteke je, da preveri, ali obstaja uporabniško ime z geslom, ki ga je vnesel obiskovalec. Podrobnosti so izpuščene, bistvo:

```
$hint = $uporabniki->aliJeUporabnikInGesloVBazi($uporabnik);
if ($hint == 0){ $response="Uporabnik ni v bazi ali pa geslo ni pravo.";
} else { //Vključimo datoteko, v kateri so funkcije za delo s sejami
    include("../../../pridelovalci/session.php");
    //Nastavimo rokovalnik seje - v parametrih klicane funkcije mu pošljemo
    // spisek funkcij iz razreda session.php. Vsaka funkcija se bo sprožila
    // ob posebnem dogodku tekom izvajanja seje. Funkcije berejo in
    // spreminjajo vrednosti tabele MySQL session.
    session_set_save_handler("sess_open", "sess_close", "sess_read",
        "sess_write", "sess_destroy", "sess_gc");
    session_start();//Začnemo izvajati sejo.
    $_SESSION['EkoKmetijeUporabnik'] = $_GET["q"]; //Določimo 1. parameter
    $_SESSION['EkoKmetijeGeslo'] = $_GET["g"]; //Določimo 1. parameter
    //Na spodnji izpis je vezan stavek IF v rokovalniku skripte
    //prijavaInRegistracija.js, od koder se je klicala poizvedba Aja
    $response=$_SESSION['EkoKmetijeUporabnik']." je prijavljen.";
}echo $response; //Vrnemo odgovor na izhod, v brskalniku se nastavi piškotek
```

Ko se koda (stran 63) izvede, spletni strežnik posreduje odgovor objektu `XMLHttpRequest`. V brskalniku se nastavi piškotek `PHPSESSID`, ki ima spremenljivki `EkoKmetijeUporabnik` in `EkoKmetijeGeslo`. Sproži se tudi funkcija JavaScript `stateChangedObstojUpImenaInGeslo`.

```
function stateChangedObstoJUpImenaInGeslo () {
    if (xmlhttpObstoJUpImenaInGeslo.readyState==4) {
        document.getElementById("txtObstoJUporabnikaVpis").innerHTML=
        xmlhttpObstoJUpImenaInGeslo.responseText;
        if(xmlhttpObstoJUpImenaInGeslo.responseText.indexOf("je prijavljen")>-1){
            var idSeje = getCookieValue("PHPSESSID");
            setTimeout(function () {window.location.href = baseDir +
                "pridelovalci/administracija.php?seja="+idSeje;}, 1000);
        } } //konec vseh stavkov if
    }
```

V primeru, da je uporabnik prijavljen, zgornja funkcija preusmeri uporabnika na administrativno spletno stran aplikacije (datoteka `administracija.php`). Ob klicu te strani se pošlje še parameter seja, ki nosi vrednost (id) seje, ki se je pred tem shranila v piškotek brskalnika. V tej skripti se naloži knjižnico `session.php` [21], nato se iz baze iz tabele `session` prebere vrstico, ki je vezana na ta id seje.

```
include("../session.php");
//unserializesession in sess_read sta v datoteki session.php
$sess_Data = unserialize(session_read($_GET['seja']));
//Iz stolpca sess_data tabele session preberemo podatek EkoKmetijeUporabnik
$upIme = $sess_Data['EkoKmetijeUporabnik'];
```

Ko skripta ugotovi kateri uporabnik je prijavljen, prikaže njegove podatke na zaslon. Zgled administrativne strani prijavljenega uporabnika prikazuje slika 56. Tu lahko administrator spreminja določene podatke in jih shrani. Ob shranjevanju se kliče skripto php, ki na podoben način ugotovi za katerega uporabnika gre in v tabeli `Ponudnik` posodobi njegove zapise.

Prednost uporabe sej je torej ta, da uporabniku ni treba ob vsaki spremembi znova vpisovati gesla. Ko se prijavi, lahko večkrat shrani spremembe ob tem, ko piše opis kmetije. Ta je formatiran v HTML.

Na koncu se uporabnik odjavi iz administracije (gumb »Odjavi« desno zgoraj). Takrat se seja konča, administrativna stran se zapre, uporabnika se preusmeri na začetno stran iskalnika.

Slika 56: Stran za administracijo predstavitvene strani

[illegible]

3.4.7 Uporabljene knjižnice in moduli

Pri izvedbi naloge so pri JavaScript in php kodi uporabljene knjižnice:

- jQuery^Z za posebne efekte prikazovanja okna aplikacije in teksta vnosnih polj, prikaz določenih vsebin strani (navodila, zemljevid, opisi logotipov) v oknih, ki se prikažejo ob kliku na specifičen element.
- PHPMailer^{AA} za pošiljanje elektronske pošte preko protokola SSL, ki zagotavlja višjo varnost komunikacije. Uporabljena je v datoteki `posljiEmailRegistracijo.php` v procesu registracije uporabnika.
- Md5 za kriptiranje gesel v podatkovni bazi. Z njeno uporabo zagotovimo, da nihče ne more vedeti gesel uporabnikov, tudi administrator ne.
- SimpleHTMLDOM^{BB} za preiskovanje elementov specifičnega tipa na spletni strani. Uporabil se je v skripti za pridobivanje podatkov z referenčnih strani (glej dodatek •).

Pri izvedbi naloge so bili uporabljeni sledeči moduli JavaScript:

- reCaptcha^{CC} – knjižnica PHP za uporabo testa CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart). Z njim se preveri, ali je odjemalec program ali človek. Uporablja se za preprečevanje zlorab prijave in registracijske forme s strani spletnih robotov. Ne želimo namreč, da bi neka programska skripta poskušala vdreti v sistem z ugibanjem gesel. Sistem je bil iznajden okoli leta 2000 in je bil sprva razvit, da bi preprečil avtomatskim programom kontinuirano ustvarjanje brezplačnih e-poštnih računov, ki bi se uporabljali za pošiljanje spama. Modul je uporabljen v modulu za prijavo in registracijo (glej podpoglavje 3.4.5.1), katerega je avtor diplomskega dela sprogramiral sam.
- JavaScript image combobox (msdropdown) – knjižnica za prikaz slik v HTML elementu `<select>`. Prednosti njene uporabe:
 - Element `<select>` ima enako obliko v različnih brskalnikih.
 - V njegova polja `<option>` lahko damo slike
 - Med `<option>` in `</option>` lahko preprosto zapišemo piko ali poljuben znak, vsebino pa določimo v parametru elementa. Na tak način pajki spletnih iskarnikov ne bodo prebrali njegove vsebine, ljudje pa jo bodo kljub temu videli. Takšne elemente uporabimo pri opcijah filtrov.
- TinyMCE^{DD} - WYSIWYG (what you see is what you get = kar vidiš, je to, kar dobiš) urejevalnik, ki bazira na HTML in JavaScript in omogoča uporabniku, da v HTML element `<textarea>` ali v druge vnaša oblikovana besedila, slike, tabele, v naprej pripravljene predloge,... Uporablja se v datoteki `administracija.php` (glej podpoglavje 3.4.6.2).

^Z jQuery: <http://jquery.com/>

^{AA} phpMailer: <http://sourceforge.net/projects/phpmailer/>

Opis delovanja SSL najdemo na strani http://sl.wikipedia.org/wiki/Secure_Sockets_Layer

^{BB} SimpleHtmlDom: <http://sourceforge.net/projects/simplehtmldom/> in <http://simplehtmldom.sourceforge.net/>

^{CC} Zadnja različica reCaptcha: <https://code.google.com/p/recaptcha/downloads/list?q=label:phplib-Latest>

^{DD} TinyMCE: www.tinymce.com

3.5 Preizkušanje aplikacije

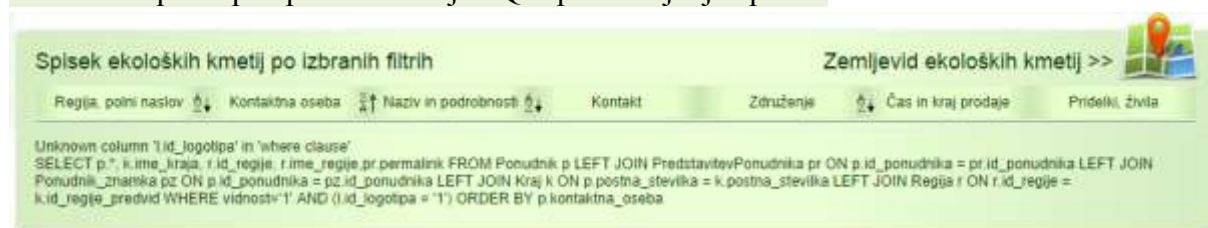
Aplikacija je bila izdelana na lokalnem računalniku z operacijskim sistemom Windows 7 in nameščenim paketom AMPPS, opisanim v začetku poglavja 3. Naknadno je bilo njeno delovanje preizkušeno na dveh strežnikih:

- Strežnik podjetja Presentia d.o.o. na paketu gostovanja za domeno boristurk.net Spletni strežnik: Cloud Linux 6.3. Strežnik http: Apache 2. Strežnik je bil opremljen z MySQL 5 in PHP 5.
- Strežnik podjetja Avant.Si d.o.o., pri katerem je bila registrirana domena s šumnikom – ekološke-kmetije.si. PHP verzija 5.4.29.MySql verzija 5.5.37.

Aplikacija preverjeno deluje v spletnih brskalnikih IE > 11.0.3, Firefox > 9.0.1, Chrome > 33.0.1750.146. Ugotovljeno je bilo, da različni brskalniki različno prikazujejo HTML element `<select>`. S tem namenom se je uporabila knjižnica Msdropdown , opisana v poglavju 3.4.7, s pomočjo katere je ta problem odpravljen.

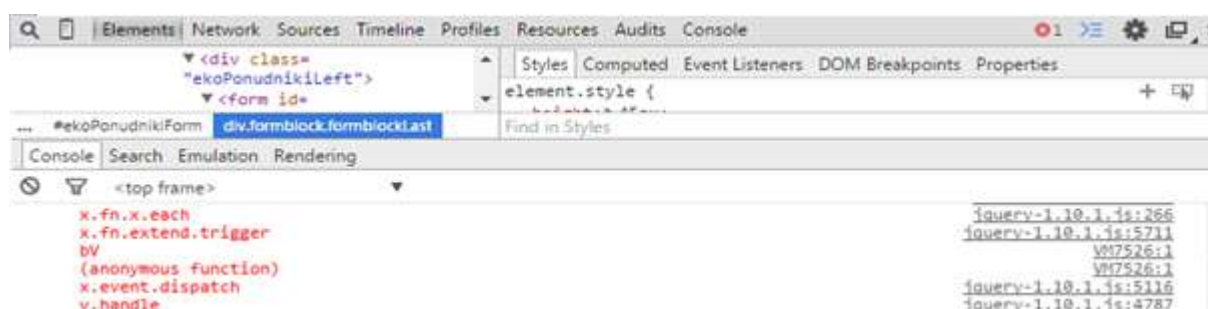
Pri izvajanju poizvedb PHP in SQL je bilo tekom testiranja urejeno, da je aplikacija morebitne napake pri uporabi iskalnika izpisala. V končni izvedbi je potrebno takšne izpise preprečiti. napake vidimo na sliki 57.

Slika 57: Izpis napak poizvedovanja SQL pri razvijanju spletne



Funkcionalnost skript JavaScript je bila preverjana sproti s pomočjo orodja Google Chrome za razvijalce (glej sliko 58).

Slika 58: Primer izpisa napake JavaScript v orodju Google Chrome za razvijalce



Uporabljena so bila še tri orodja za preverjanje vsebine spletne strani na različne načine. Pri vseh orodjih se v vnosno polje vpiše naslov spletne strani (ekološke-kmetije.si oziroma xn--ekoloke-kmetije-96c.si^{EE}, ker je domena s šumnikom in se pretvori v ACE (ASCII Compatible Encoding) niz)) in požene poizvedbo, ki vrne specifično analizo za spletno stran. Ocene orodij prikazujemo preko naslednjih slik: 59, 60, 61.

^{EE} Arnes Register: Pogosta vprašanja: <http://www.register.si/faq/idn.html>

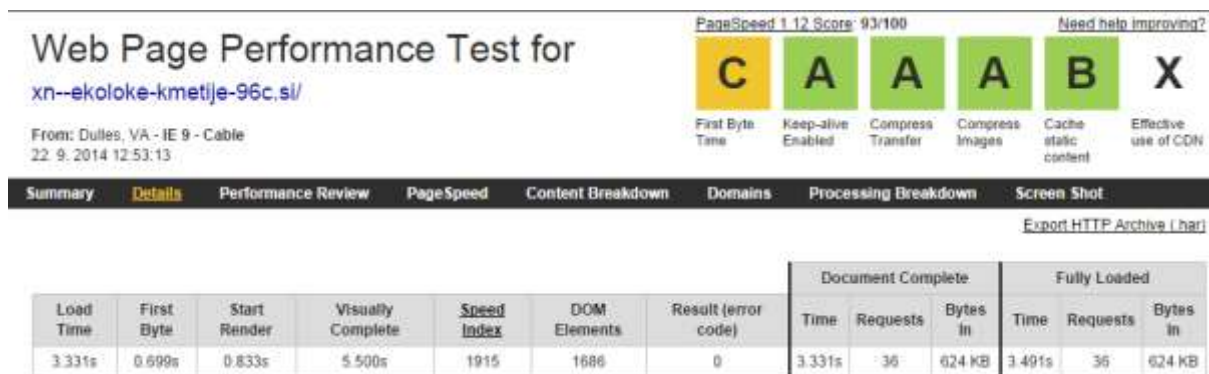
Stran je validirana s strani organizacije W3C^{FF}, ki je zadolžena za njuno standardizacijo.

Slika 59: Orodje W3C ocenilo, da je ima iskalnik ekoloških kmetij ustrezno kodo HTML 5.

This document was successfully checked as HTML5!	
Result:	Passed, 1 warning(s)
Address :	<input type="text" value="http://xn--ekoloke-kmetije-96c.si/"/>
Encoding :	utf-8 <input type="button" value="(detect automatically)"/>
Doctype :	HTML5 <input type="button" value="(detect automatically)"/>
Root Element:	html
Root Namespace:	http://www.w3.org/1999/xhtml

Za potrebe optimizacije hitrosti nalaganja vsebine strani je bilo uporabljeno orodje webpagetest^{GG}. Po ocenah sodeč je problematičen samo zapoznel odziv strežnika, na kar preko aplikacije ne moremo vplivati.

Slika 60: orodje webpagetest je ocenilo hitrost strani z oceno 93/100.



Orodje za analizo ključnih besed webseoanalytics^{HH} - je izpisalo rezultate za našo stran.

Optimizacija SEO je uspešna, vse ključne besede so v zelenem območju, nobena ni prepoznana kot možen spam, vse so na strani zastopane v ustrezni gostoti.

Slika 61: Orodje webseoanalytics je ocenilo, so ključne besede na strani v ustrezni gostoti.

KEYWORD ANALYSIS				
SINGLE KEYWORDS	KEYWORDRANK	OCCURRENCES	DENSITY	POSSIBLE SPAM
ekološki		15	1.68%	
kmetije		12	1.35%	
ekološke		10	1.12%	
kmetij		12	1.35%	
živil		10	1.12%	
SHOW MORE				
2 WORD PHRASES	KEYWORDRANK	OCCURRENCES	DENSITY	POSSIBLE SPAM
ekološke kmetije		6	0.67%	
ekoloških kmetij		8	0.9%	
ekoloških živil		2	0.22%	
iskalnik ekoloških		3	0.34%	
zastopnike eko.		2	0.22%	

^{FF} <http://validator.w3.org/>.

^{GG} Hitrost merjena pri na strani <http://www.webpagetest.org>

^{HH} Storitve SEO na <http://www.webseoanalytics.com/free/seo-tools/page-analyzer-keyword-density-tool.php>

3.6 Primer skripte PHP za primerjanje podatkov v bazi s tujimi podatki

Nekatere dodatne podatke o kmetijah lahko pridobimo z referenčnih strani blagovnih znamk. Na primer s strani www.eko-kmetije.info. Na tej strani je iskalnik kmetij, ki izpiše rezultate iskanj na več podstraneh. Pričakujemo, da se bodo nekateri kontaktni podatki kmetij spreminjali in da bodo s strani te organizacije morda ažurni pri posodabljanju teh podatkov. Zato je bila sprogramirana skripta PHP `updatePonudniki.php`, ki zna poiskati in obdelati podatke vseh rezultatov tega iskalnika.

Skripta rekurzivno obišče vse podstrani, iz njih pobere pomembne podatke ter jih primerno obdeli (izpiše na zaslon, obarva spremembe, shrani v bazo). Pri tem je bila uporabljena knjižnica SimpleHTMLDOM (glej podpoglavje 3.4.7), ki omogoča, da v neko spremenljivko naložimo vsebino HTML, nato pa se sklicujemo na specifične elemente in vračamo njihove vrednosti. Primer:

```
//V spremenljivko $html naložimo vsebino referenčne spletne strani
$html = file_get_html('http://www.eko-kmetije.info/...');
$myTable = $html->find('table',3); //Poiščemo neko tabelo na tej strani
//V zanki obhodimo vse vrstice tabele
foreach($myTable->find('tr') as $row) {
    $stolpec = $row->find('td',0)->plaintext; //Poiščemo prvi stolpec
    $vrednost = $row->find('td',1)->plaintext; //Poiščemo drugi stolpec
    $table[$stolpec] = $vrednost; //Priredimo vrednost določenemu stolpcu
}
echo '<pre>';print_r($table);echo ($table["Spletna stran"]);echo '</pre>';
```

Zgoraj omenjeni iskalnik izpiše rezultate v obliki tabele HTML (`<table>`). Izvedba zgornje kode izpiše elemente te tabele. Primer izpisa je prikazan na sliki 62. Skripta `updatePonudniki.php` način takšnega dostopa do elementov uporablja pri svojem delovanju:

- 1) Rekurzivno se poveže na vse podstrani rezultatov iskanj iskalnika www.eko-kmetije.info.
- 2) Na vsaki podstrani prebere tabelo rezultatov.
- 3) Sprehodi se skozi vse vrstice te tabele.
- 4) Ob prehodu skozi vrstice primerja vse vrednosti po stolpcih z vrednostjo pripadajočega podatka v naši bazi `Ponudnik`. Sproti izpisuje (tudi shranjuje) podatke.

Tako preverimo (tudi ažuriramo) ali je pri specifični kmetiji prišlo do spremembe določenega podatka.

```
Array
(
    [ ] =>
    [ ] =>
    [Bogataj Dušan] => Status:
    [Domače ime] => Pr" Jureč
    [Naslov] => Ledine 19
    [Pošta] => 5281 Spodnja Idrija
    [Telefon] => 05/377-90-96
    [GSM] => 041-919-473
    [Fax] =>
    [E-pošta] => pr-jurec@volja.net
    [Spletna stran] => http://www.p
    [Članstvo v združenjih] =>
    [Čas in kraj prodaje] =>
    [Opis poti do kmetije] => Logat
    [Posebnosti] => Etnološka zbirka
    [Pridelki, živila] => Blitva, B
    [Dodatna ponudba] => Kruh (pšen
    [Nazaj na rezultate] =>
)
```

Slika 62: Izpis elementov spremenljivke z vsebino HTML.

Na naslednji strani je primer okrnjenega dela skripte `updatePonudniki.php`, kjer se preverja le podatke o spremembi GSM. Slika 63 prikazuje izpis te skripte.

Id ponudnika	Kontaktna oseba	GSM
1	Adamič Miran	041-624-639Test ==> 041-624-639
2	Ambrožič Lidija	031 - 410 - 506
3	Anžih Marko	040 - 768 - 931
4	Arnež Jože	041- 970 - 823
5	Artičoka Ksenija Verbič s.p.	041 - 673 - 766
6	Ažnroh Anton	041 - 294 - 804
7	Baligač Jožef	
8	Bergant Stanislav	040-626-249
9	Berne Anton	
10	Bočaj Janko	041 - 607 - 520

Slika 63: Primerjava podatkov tujega iskalnika s podatki iz naše baze. Če so podatki različni, se obarvajo z rdečo.

```

<?php //Okrnjena skripta updatePonudniki.php
require('./simple_html_dom.php'); //Knjižnica omenjena v poglavju
//Vključimo datoteko za nastavitev strežnika, baze, jezika
include("../../class.StreznikInBaza.php");?>
<html><head>//Za lepši izpis strani in prikaz šumnikov
<title>Posodobitev podatkov ekoloških kmetij</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<link href="../../css/style.css" rel="stylesheet" type="text/css" />
</head><body><?php
    $setupDB1 = vrniKonfiguriranoBazo();
    $query="SELECT p.*, k.ime_kraja, r.ime_regije
            FROM Ponudnik p
            LEFT JOIN Kraj k on p.postna_stevilka = k.postna_stevilka
            LEFT JOIN Regija r on r.id_regije = k.id_regije
            ORDER BY id_ponudnika limit 10";//Preverjali bomo prvih 10 ponudnikov
    echo("<table class=\"ponudniki\" cell-spacing=0 cell-padding=0>");
    echo("<thead> <tr>
        <th scope=\"col\">Id ponudnika</th>
        <th scope=\"col\">Kontaktna oseba</th><th scope=\"col\">GSM</th>
    </tr> </thead> ");
    if ($ponudniki = mysql_query($query)){
        while($row = mysql_fetch_assoc($ponudniki)) {
            $referencniUrl = $row["referencni_url"];
            if (strlen($referencniUrl) == 0 or is_null($referencniUrl)){
                $referencniUrl = "null"; }
            $table = array();
            if (strpos($row["referencni_url"], 'www.eko-kmetije.info') == false) {
                continue; } //Obravnavamo reference s točno določenega spletišča
            //Uporabimo rešitev
            $html = file_get_html($row["referencni_url"]);
            $myTable = $html->find('table',3);
            foreach($myTable->find('tr') as $rowTr) {
                $stolpec = $rowTr->find('td',0)->plaintext;
                $vrednost = $rowTr->find('td',1)->plaintext;
                $table[$stolpec] = $vrednost;
            }
            $rowOddEven = "soda";//Za lepši izpis sode vrstice izpišemo drugače
            if ($row["id_ponudnika"] % 2) {$rowOddEven="soda";}
            else{$rowOddEven="liha";}
            echo("<tr class='". $rowOddEven. "'>");
            echo("<td class=\"cell brezSprememb\">". $row["id_ponudnika"] . "</td>");
            echo("<td class=\"cell brezSprememb\">". $row["kontaktna_oseba"] . "</td> ");
            if ($row["gsm"] == $table["GSM"] ) {
                echo("<td class=\"cell brezSprememb\">". $row["gsm"] . "</td>");
            } else {
                echo("<td class=\"cell novo\">". $row["gsm"] . " ==> " . $table["GSM"] . "</td>"); }
            echo("</tr> ");
        }
    }else{echo (mysql_error ()); } //Izpis morebitne napake
    echo("</table>");
    mysql_close($setupDB1->vrniPovezavo());//Zapremo povezavo z bazo MySql
?>
</body></html>

```

4 Sklep in nadaljnje delo

4.1 Sklepne ugotovitve

V obdobju pisanja diplomske naloge sem si ogledal nekaj dokumentarcev o mravljah in me je povsem navdušila njihova sposobnost organizacije. So zelo uspešne množice insektov, ogromno jih je, toliko, da v mravljiščih plezajo ena po drugi. Njihov uspeh ne leži v njihovih sposobnostih ali moči posameznikov, ampak v organizaciji, koordinaciji njihove združbe. Kar rabimo ljudje za boljšo simbiozo v ekosistemu, ki se v zadnjih letih zaradi nekorektnosti v naših dejanjih ruši, je posnemanje teh malih bitij - boljša ekološka organizacija med ponudniki in potrošniki, ter večja koordinacija v ekološkem osveščanju. Aplikacija je prispevek družbi, ki to dvojje izboljšuje.

Eden najboljših načinov kako zmanjšati prodajo hrane, prežete z ostanki pesticidov (v trgovinah lahko na policah konvencionalno pridelanih citrusov opazimo napise, da olupki niso užitni... ekološki citrusi pa so užitni v celoti), je, da te hrane ne kupimo. Seveda pa moramo najprej imeti alternativo. Razvita spletna aplikacija služi temu, da bo alternativna stran močnejša.

Tako kot se v kognitivno znanost povezujejo filozofija, psihologija, računalništvo, nevroznanost, lingvistika in družbene vede, da bi raziskovali duševne procese, bi bilo dobro, da se ljudje iz različnih disciplin povežemo v iskanju učinkovitih rešitev za vzpostavitev ekološkega načina kmetovanja. Idej še ni zmanjkalo.

4.2 Nadaljnji razvoj

- V načrtu je možnost interakcije neregistriranih obiskovalcev. Obiskovalci bodo v prihodnosti imeli možnost pohvaliti kvaliteto specifičnega ponudnika. Princip bo podoben, kot ga ima Facebook Like ali Google +.
- Dodatno bodo lahko obiskovalci opozorili administratorja, če bodo mnenja, da ponudnik krši zahteve, ki jih je potrebno izpolnjevati za pridobitev in ohranjanje ekološkega certifikata. V primeru, da bi prišlo do več prijav, lahko administrator opozori inšpekcijsko službo. Vzpostavil bi se lahko tudi sistem za evidentiranje (recimo s posebno ikono) takšnih kmetij v rezultatih iskanj.
- Razviti prilagojeno različico za telefone z majhno resolucijo zaslona.
- V načrtu so tudi letaki v obliki zloženke, ki bodo ohranjali skladnost s CGP aplikacije.
- Za kriptiranje gesel uporabiti CriptoJS^{II}, AES algoritem s podporo pbkdf2^{JJ, KK},

^{II} Knjižnica CriptoJS je dostopna na strani <https://code.google.com/p/crypto-js/>.

^{JJ} Demonstracija algoritma pbkdf2 je na stran <http://anandam.name/pbkdf2/>.

^{KK} Razlog zakaj ne uporabljati kriptiranja md5 je podan na strani <https://crackstation.net/hashing-security.htm>.

Literatura

- [1] STELE, Aleš: O kmetijstvu doma in drugje po EU [Elektronski vir] / [avtorja Aleš Stele in Irena Žaucer]. - El. knjiga. - Ljubljana : Statistični urad Republike Slovenije, 2013. - (Zbirka Brošure / Statistični urad Republike Slovenije). Dostopno na:
http://www.stat.si/doc/pub/kmetijstvo_EU.pdf (25.9.2014)
- [2] Statistični urad RS: Pomembnejši podatki Popisa kmetijstva: delovna sila na kmetijskih gospodarstvih, Slovenija, 2010 - začasni podatki. Dostopno na:
http://www.stat.si/novica_prikazi.aspx?id=4241 (25.9.2014)
- [3] Ministrstvo za kmetijstvo, gozdarstvo in prehrano RS. 2010. Izhodišča Slovenije za razpravo os skupni kmetijski politiki EU po letu 2012. Dostopno na:
http://www.mko.gov.si/fileadmin/mko.gov.si/pageuploads/podrocja/SKP/Izhodisca_SI_o_SKP_po_2013.pdf (25.9.2014)
- [4] Ministrstvo za kmetijstvo in okolje: Evidenca certifikatov ekološkega kmetovanja. Dostopno na:
<https://webapl.mkgp.gov.si/bioWeb/informacije.php> (25.9.2014)
- [5] Inštitut za kontrolo in certifikacijo v kmetijstvu in gozdarstvu: Vodnik po ekoloških kmetijah. Dostopno na:
<http://www.eko-kmetije.info/iskanje.asp> (25.9.2014)
- [6] Repovš, Jernej. 1995. Kako nastaja in deluje učinkovita, tržno usmerjena celostna grafična podoba kot del simbolnega identitetnega sistema organizacij. Stran 77. Ljubljana: Studio Marketing.
- [7] Koleč, Tomaž. 2000. Kako do dobrega imena in simbola. Podjetnik. Dostopno na:
<http://www.podjetnik.si/clanek/kako-do-dobrega-imena-in-simbola-20001008> (25.9.2014)
- [8] Skrt, Radoš. 2009. Kako do top pozicij na iskalnikih? Seminar. Oglas seminarja dostopen na:
<http://seminarji.nasvet.com/optimizacija-strani-seminar.php> (25.9.2014)
- [9] Spletnik, spletne tehnologije d.o.o.: Kako spletni uporabniki iščejo s pomočjo iskalnikov. Dostopno na:
<http://www.optimizacija-strani.com/seo-prirocnik/kako-spletni-uporabniki-iscejo-s-pomocjo-iskalnikov.html> (25.9.2014)
- [10] Wikipedija: spletni pajek. Dostopno na:
http://en.wikipedia.org/wiki/Web_crawler (25.9.2014)
- [11] Google: Googlov začetni vodnik za optimizacijo za iskalne mehanizme. Različica 1.1 z dne 13. novembra 2008. Dostopno na:
<http://static.googleusercontent.com/media/www.google.si/sl/si/intl/sl/webmasters/docs/search-engine-optimization-starter-guide-sl.pdf> (25.9.2014)
- [12] Najdi.si: Optimizacija spletnih strani. Dostopno na:
<http://pomoc.najdi.si/za-lastnike-spletnih-strani/> (25.9.2014)

- [13] Checkupdown: Errors reported by CheckUpDown. Dostopno na:
www.checkupdown.com/status/error1.html (25.9.2014)
- [14] w3school: HTML4 and HTML5 Tutorial. Dostopno na:
<http://www.w3schools.com/html/default.asp> (25.9.2014)
- [15] w3school: HTML Head. Dostopno na:
http://www.w3schools.com/html/html_head.asp (25.9.2014)
- [16] Elektronsko gradivo. Tehnološki šolski center Kranj: Podatkovne baze - Modeliranje - Relacijski model. Dostopno na:
<http://drenovec.tsckr.si/> in <http://drenovec.tsckr.si/model/relac.htm> (25.9.2014)
- [17] Elektronsko gradivo. Šolski center Novo mesto: Podatkovne zbirke. Dostopno na:
http://www.ecnm.si/e-gradivo/BAZE/primarni_klju.html (25.9.2014)
- [18] MySQL: How MySQL Uses Indexes. Dostopno na:
<http://dev.mysql.com/doc/refman/5.5/en/mysql-indexes.html> (25.9.2014)
- [19] w3school: The XMLHttpRequest Object . Dostopno na:
http://www.w3schools.com/xml/xml_http.asp (25.9.2014)
- [20] Google Maps JavaScript API v3: Geocoding service. Dostopno na:
<https://developers.google.com/maps/documentation/javascript/> (25.9.2014)
- [21] Tuxradar: Cookies and sessions, Using Sessions, Files vs. Databases. Dostopno na:
www.tuxradar.com/practicalphp/10/3/7 (25.9.2014)

Dodatki

1 Kaj je ekološka pridelava?

V podpoglavju 2.2.1 je bilo zapisano, da je za to, da bi aplikacija koristila večjemu krogu ljudi, potrebno osveščati ljudi o razlikah med konvencionalnim in ekološkim načinom pridelave ter jim predstaviti bistvene prednosti ekološkega načina pridelave, ki niso pomembne zgolj za človeka, pač pa za celoten ekosistem in s tem za bodoče generacije ljudi.

»Ekološka pridelava je celotni sistem upravljanja kmetijskega gospodarstva in pridelave hrane, ki združuje najboljšo okoljsko prakso, visoko raven biotske raznovrstnosti, ohranjanje naravnih virov, uporabo visokih standardov dobrega počutja živali in način pridelave v skladu s preferencami nekaterih potrošnikov za proizvode, pridelane z uporabo naravnih snovi in postopkov. Postopek ekološke pridelave igra tako dvojno družbeno vlogo: po eni strani oskrbuje specifičen trg in s tem zadošča povpraševanju potrošnikov po ekoloških proizvodih, po drugi strani pa zagotavlja javne dobrine in s tem prispeva k varovanju okolja, dobremu počutju živali in razvoju podeželja.« [1d]

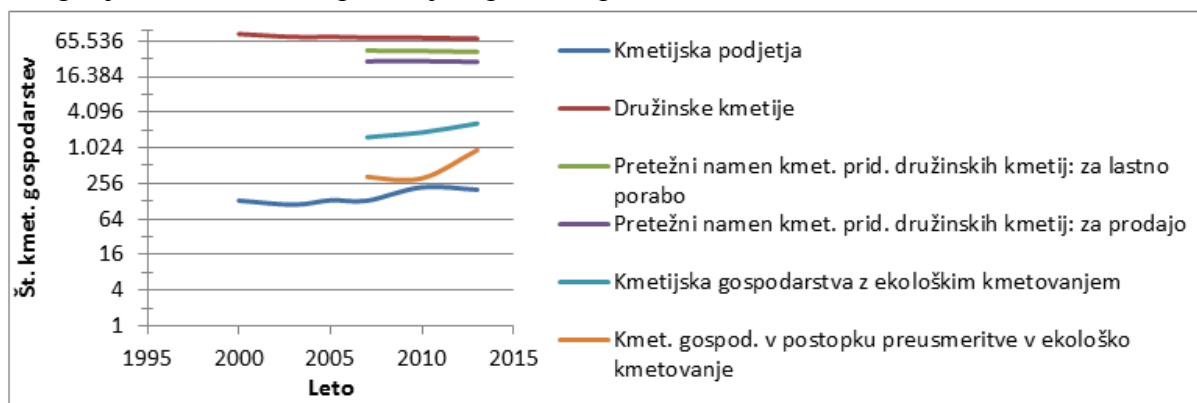
2 Kmetijstvo v Sloveniji

Eden od ciljev aplikacije je vplivati na povečanje števila ekoloških pridelovalcev živil. Zakaj je to za Slovenijo zelo pomembno, si lahko ogledamo skozi pregled nad trenutnim stanjem kmetijskega gospodarstva in prehranske varnosti v splošnem. »Vsaka ekološka kmetija šteje!«

Da bi se prepričali o tem, kakšna je situacija in zakaj je pomembno promovirati samooskrbo ter ekološko kmetovanje, potrebujemo kar nekaj podatkov, ki pa se jih žal ne da predstaviti na eni strani.

V Sloveniji v zadnjem desetletju število kmetijskih gospodarstev upada, število ekoloških kmetijskih gospodarstev pa narašča. V naslednjih poglavjih bo predstavljeno, kaj je samooskrba, prehranska varnost, kako je pomembna in kakšne posledice lahko prinese pre nizka prehranska varnost.

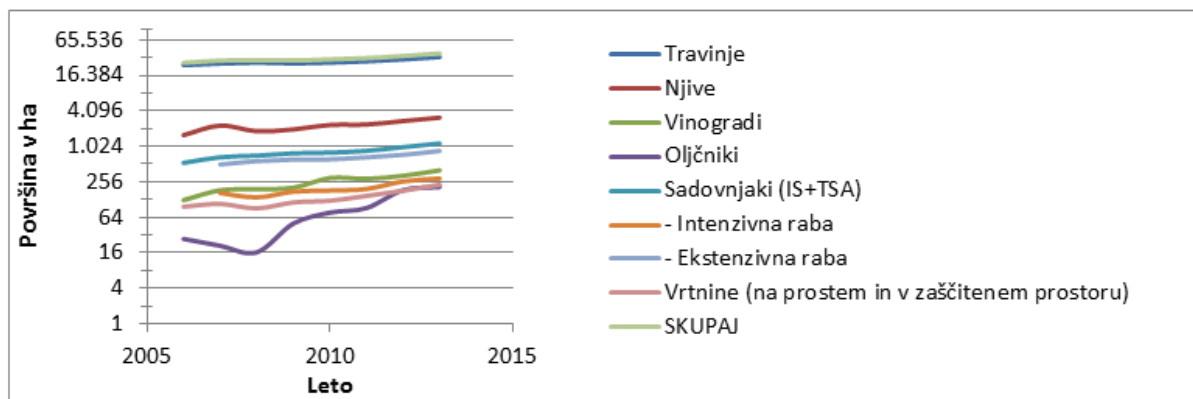
Statistični urad na strani [2d] ponuja možnost izbire podatkov glede na leta in različne kategorije. S strani so bili pridobljeni podatki, prikazani na sliki dodatka 1.



Slika dodatka 1: Graf števila kmetijskih gospodarstev po letih od leta 1995 naprej

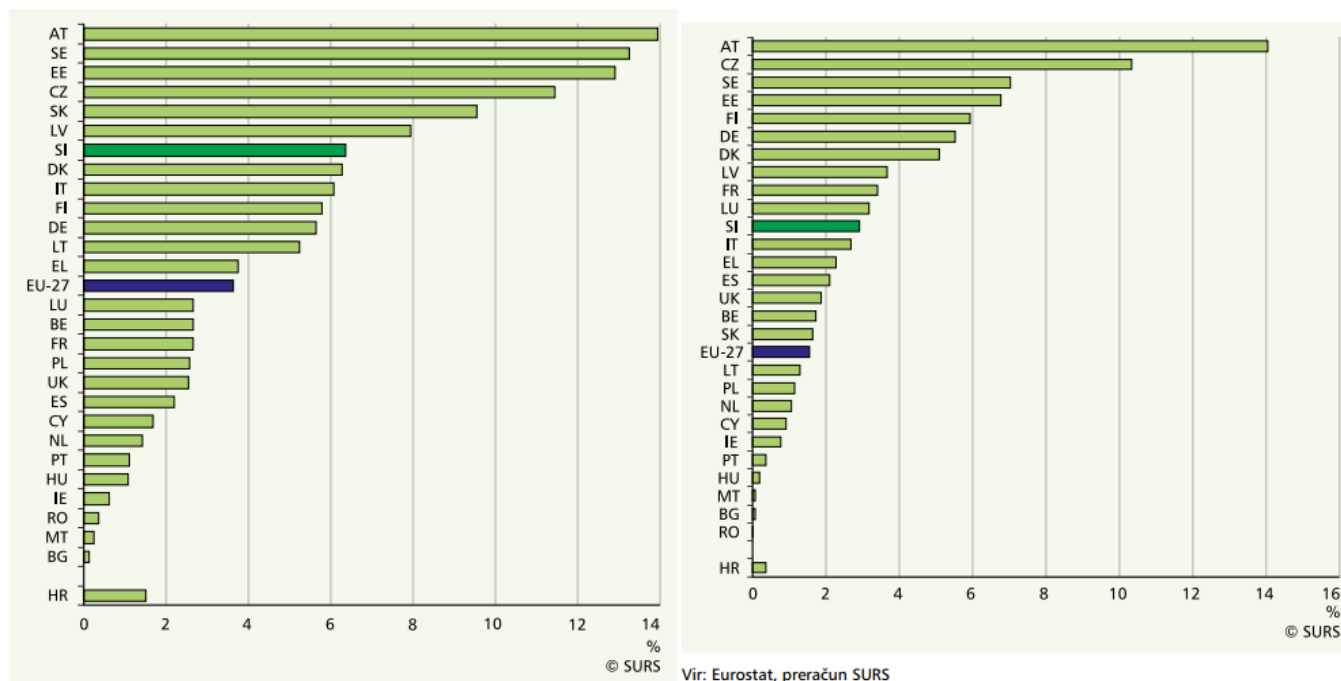
2.1 Analiza stanja ekološkega kmetovanja

»V letu 2013 je bilo v ekološko kontrolo vključenih 3.049 kmetijskih gospodarstev (kar predstavlja 4,1% vseh kmetij v Sloveniji) z 38.664,49 ha kmetijskih zemljišč v uporabi (8,4% od vseh kmetijskih zemljišč v uporabi v letu 2013). Od teh je 2.232 kmetij že zaključilo preusmeritveno obdobje (pridobilo eko certifikat), ki traja najmanj 24 mesecev od prve prijave v kontrolo.« [3d]



Slika dodatka 2: Površina kmetijskih zemljišč po letih za več kategorij

V [3d] piše, kakšna je bila struktura in obseg (v ha) obdelovalnih površin, vključenih v kontrolo ekološke pridelave. Izvleček: »V Sloveniji so pašniki in travniki zavzemali okoli 69 % vseh ekološko obdelanih zemljišč, žita 4 %, sadovnjaki in jagodičje po okoli 3 % in vinogradi 1 %«. Na osnovi njihovih podatkov je bila izdelan graf, prikazan na sliki dodatka 2. S strani Statističnega urada [4d] dobimo 2 grafikona, ki sta prikazana na sliki dodatka 3.



Vir: Eurostat, preračun SURS

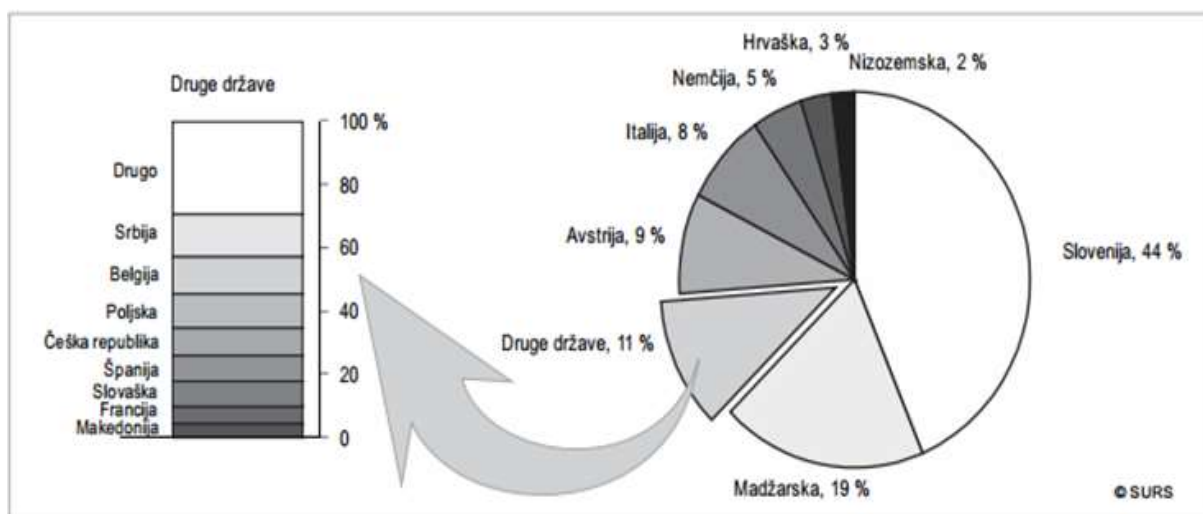
Vir: Eurostat, preračun SURS

Slika dodatka 3: Grafikon, ki prikazuje odstotek kmetijskih gospodarstev z ekološkim načinom kmetovanja v Evropi in Hrvaški za leto 2010 (levo) in grafikon, ki prikazuje ekološko obdelovana KZU v EU-27 in Hrvaški za leto 2010

2.2 Prehranska varnost in samooskrba

»Prehranska varnost obstaja v primeru, ko imajo vsi prebivalci vedno fizični in ekonomski pristop do zadovoljivih količin zdrave hrane, ki omogoča zadovoljevanje prehranskih potreb za zdravo in aktivno življenje« [5d]

Prehranska varnost je izrazito strateški pojem. Država bi morala poskrbeti, da bi bili v primeru krize sposobni sami zadovoljiti vsaj osnovne potrebe vseh prebivalcev. Spodaj bo pojasnjeno, da Slovenija trenutno tega ni sposobna. Primanjkljaj kompenziramo z uvozom hrane iz tujine, ampak mednarodni prehranski trg ni nekaj, na kar bi se lahko zanašali, še zlasti ne zaradi nenehnega povečevanja števila prebivalstva. Izvor hrane, potrošene v Sloveniji, si lahko ogledamo na grafu, ki ga prikazuje slika dodatka 4.



Slika dodatka 4: Izvor hrane, potrošene v Sloveniji

Posledica prenizke prehranske varnosti je lahko lakota, ta pa lahko prinese kup drugih problemov. Po drugi strani pa je manjša prehranska samooskrba neposredno povezana z manjšim številom zaposlenih in manjšim bruto domačim proizvodom.

Plut [5d] navaja, da bi po virih, ki jih navaja, za varno prehransko oskrbo v Sloveniji potrebovali na prebivalca najmanj 2500 - 3000 m² kmetijskih zemljišč v uporabi. To je 0,25 ha – 0,3 ha na prebivalca. Poglejmo si tabelo, pridobljeno iz podatkov statističnega letopisa 2012 [6d] in skupaj ugotovimo, ali dosegamo ta minimum.

Leto	Kmetijska zemljišča v ha	Skupaj prebivalcev	Državna površina v km ²	Delež kmečkega prebivalstva v %	Št. hektarov kmetijskih zemljišč na prebivalca
1948	1003712	1391873	19981	49,0	0,72
1971	921201	1727137	20273	20,4	0,53
1981	888667	1891864	20273	9,2	0,47
1991	561294	1913355	20273	7,6	0,29
2002	509624	1964036	20273	-	0,26
2011	458214	2050189	20273	-	0,22

Iz popisa kmetijstva leta 2010 [7d]:

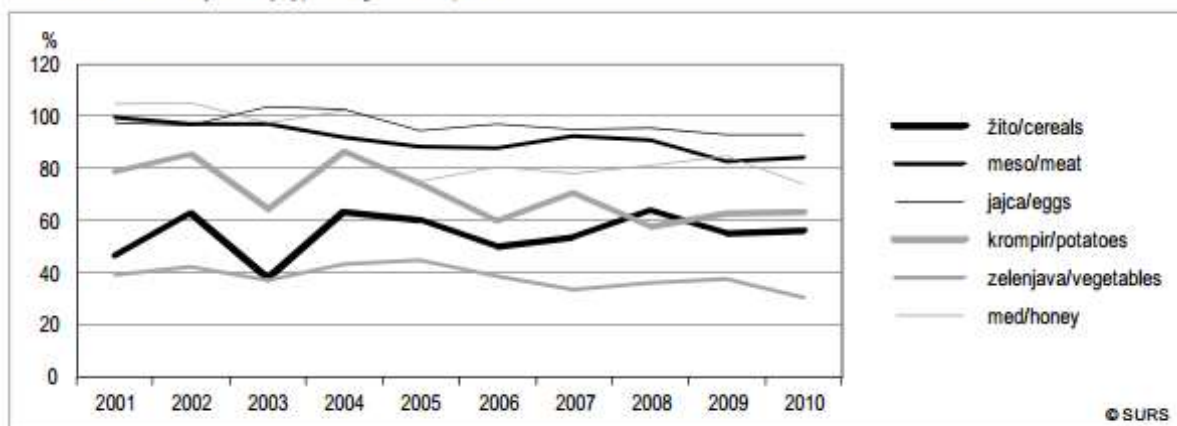
- »Po končnih podatkih iz Popisa kmetijstva 2010 smo imeli v Sloveniji 74.646 kmetijskih gospodarstev, kar je skoraj 14 % manj kot pred desetimi leti;...«
- »Skupno število delovno aktivnih oseb v kmetijstvu se je v zadnjem desetletju zmanjšalo za 19 %«
- »V letu 2010 so redili živino na 58.648 kmetijskih gospodarstvih. Z rejo živine se je od leta 2000 prenehalo ukvarjati kar dobrih 18.800 kmetijskih gospodarstev«

S strani Statističnega urada [8d] izvemo, da se je v letu 2013 s kmetijsko pridelavo ukvarjalo za 3 % manj kmetijskih gospodarstev kot v letu 2010, vendar so bila ta večja. Podatki nakazujejo, da v Sloveniji z obstoječimi prehranjevalnimi navadami in obstoječo prakso kmetovanja nimamo več dovolj kmetijskih zemljišč v uporabi za zagotovitev prehranske varnosti.

Na Statističnem uradu [9d] je na voljo bilanca proizvodnje in potrošnje kmetijskih proizvodov za Slovenijo leta 2010. Iz podatkov razberemo, da imamo od leta 2004 dalje stopnjo samooskrbe manjšo od 100% pri pridelavi žit, mesa, jajc, krompirja, zelenjave in medu. Splošno stanje samooskrbe prikazuje slika dodatka 5.

Grafikon 1: Stopnja samooskrbe po vrstah kmetijskih proizvodov, Slovenija

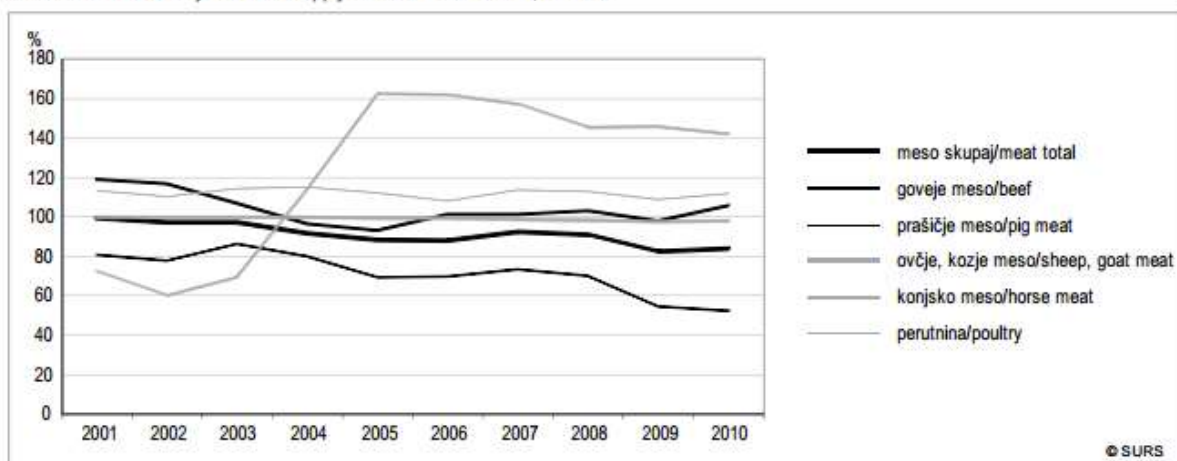
Chart 1: Self-sufficiency rate by type of agricultural products, Slovenia



Vir/Source: SURS/SORS

Grafikon 13: Stopnja samooskrbe v bilanci za meso, Slovenija

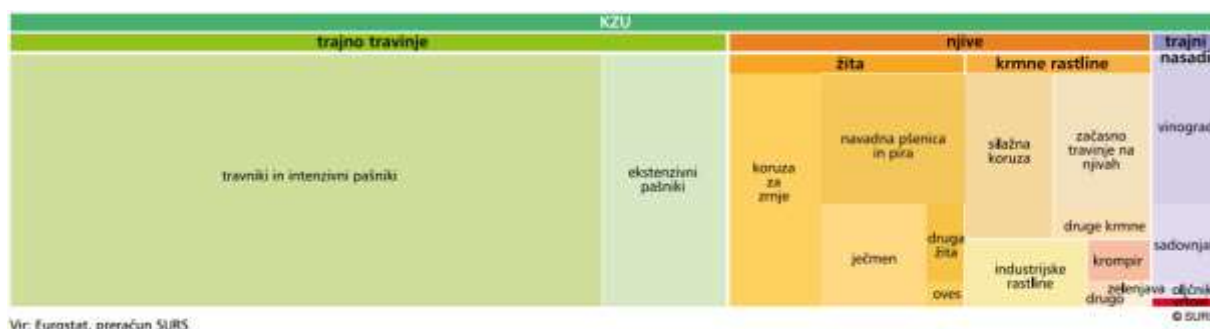
Chart 13: Self-sufficiency rate in the supply balance sheet for meat, Slovenia



Vir/Source: SURS/SORS

Slika dodatka 5: Grafikon stopnje samooskrbe po vrstah kmetijskih proizvodov (zgoraj) in grafikon stopnje samooskrbe v bilanci za meso (spodaj).

Da bi vedeli, kakšen pomen imata zgornja dva grafa, si na sliki dodatka 6 oglejmo, kakšna je bila raba KZU na kmetijskih gospodarstvih leta 2010 [10d].



Slika dodatka 6: Raba KZU na kmetijskih gospodarstvih leta 2010

Skratka, na preveč področjih imamo prenizko samooskrbo, kar je slabo za državno blagajno. Denar za stvari, ki bi jih lahko proizvedli doma, se steka v tujino, doma pa narašča število brezposelnih, ki bi se lahko ukvarjali s pridelavo hrane.

3 Zakaj število kmetijskih površin upada

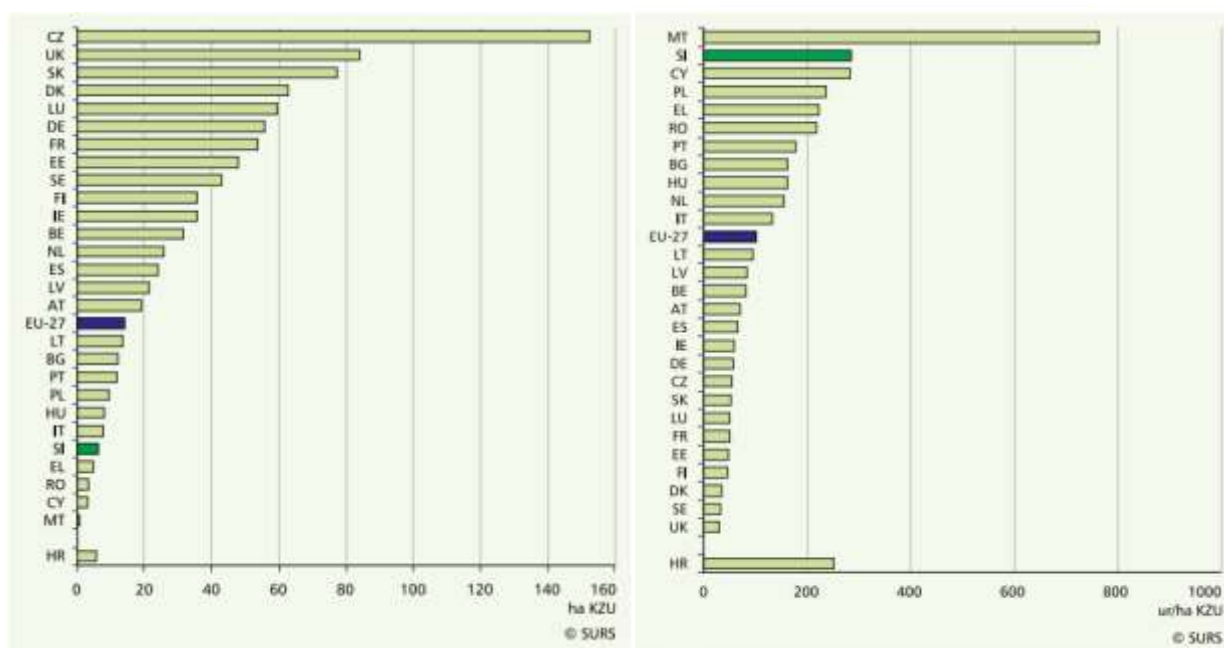
Odgovor na vprašanje, zakaj je število kmetijskih površin upadalo, ko je prišlo do industrijskih revolucij in modernizacije, uporabe boljših orodij, je jasen – na manjši površini se je pridelalo več hrane. Težje pa je odgovoriti, zakaj število kmetijskih površin upada, ko imamo na račun tega manj zaposlenih, slabše gospodarske razmere, in se postavljamo v prehrambno nevarnost v primeru krize.

Na splošno lahko glede tega zgolj ugibamo, čeravno so zadeve nekako očitne:

- Spreminja se namembnost kmetijskih površin. Na ozemlju Slovenije se delež gozdnih površin povečuje že več kot sto let. Na račun tega se zmanjšuje št. kmetijskih površin. Podatki v tabeli so iz vira [12d]
- Birokracija stvari otežuje. Kdor se želi začeti ukvarjati s kmetijstvom, se mora na primer ukvarjati s kompliciranimi postopki za pridobitev dovoljenja za postavitev večje tople grede ali hleva za živino, objekta za spravlanje sena ipd. Postopki se znajo na občinah včasih po nepotrebnem zavleči za več let. Država bi morala vzpodbujati tiste, ki želijo delati, ne pa jih omejevati s smešno zakonodajo in dolgimi postopki.
- Hrana v trgovinah je velikokrat cenejša in prihaja iz tujine. Iz podatkov o kmetijskih gospodarstvih po velikostnih razredih kmetijskih zemljišč v uporabi [11d] lahko izračunamo, da je leta 2010 s površinami nad 10 ha upravljalo zgolj dobrih 15% kmetijskih gospodarstev. Večja in bogatejša kmetijska gospodarstva si lahko privoščijo mehanizme, na račun katerih imajo s pridelavo živil manj stroškov, oziroma vložijo manj časa na enoto obdelovalne površine. Posledica tega je, da lahko živila prodajajo po manjši ceni. Kaj hitro lahko ugotovimo, da povprečno slovensko kmetijsko gospodarstvo ne zmore več konkurirati cenam na trgu. O tem pričata tudi grafikona s slike dodatka 7.

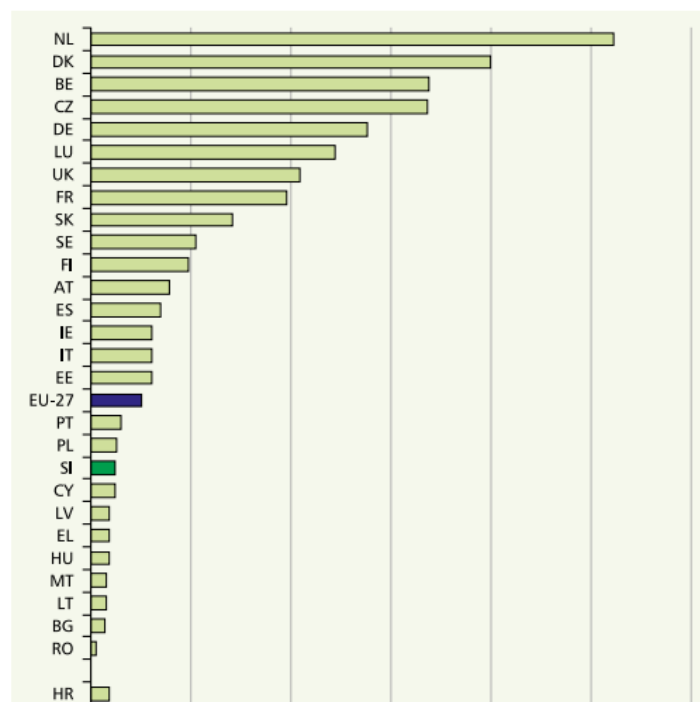
Leto	gozdne površine v %
1875	35
1947	43
1990	53
2009	60

Razberemo lahko, da je povprečno slovensko KMG v povprečju več kot pol manjše od povprečnega KMG v celotni EU-27.



Slika dodatka 7: Grafikon o povprečni količini vloženega dela na 1 ha KZU po državah za leto 2010 (levo) in grafikon o povprečni velikosti kmetijskega gospodarstva (KMG) po površini KZU leta 2010 (desno).

Na strani je še grafikon (glej sliko dodatka 8) o povprečnem standardnem prihodku na KMG za leto 2010. Iz podatkov se lahko sami prepričamo, da v Sloveniji kmetijska gospodarstva vlagajo v pridelovanje živil več časa kot drugod in imajo precej nižek standardni prihodek na KMG.



Slika dodatka 8: Grafikon o povprečnem standardnem prihodku na KMG za leto 2010

Da bi povečali št. kmetijskih površin, ni nujno, da se poveča velikost kmetijskih gospodarstev. Lahko se poveča število kmetijskih gospodarstev. Potrebno je najti način, kako spodbuditi potrošnike k nakupu lokalnih živil in kako znižati cene teh živil.

En od načinov je boljša organizacija – sistem, ki poveže potrošnike in pridelovalce. Tak sistem je na primer aplikacija. Ukrepi bi seveda lahko prišli tudi s strani vlade: večje subvencije, hitrejši postopki s strani birokracije, nenazadnje zakonska omejitev uvoza v primeru, ko država nima več prehranske varnosti.

S tem, ko supermarketi odkupujejo živila v tujini, so sami finančno na boljšem položaju – prodajo več in plačajo manj. Očitno to ustreza tudi potrošnikom, ki zaradi slabih finančnih razmer, prihrankov in neosveščenosti o pomembnosti samooskrbe, v supermarketih raje kupujejo cenejša živila iz tujine, kot nekoliko dražja lokalna. S tem je krog sklenjen, velike trgovine prodajajo čedalje več živil iz tujine.

3.1 Kako zagotoviti prehransko varnost?

Zelo dobra poteza bi bila, če bi prišel kakšen ukrep s strani politike. Ampak glede na to, da so problemi politikom znani že več let, in da število kmetijskih gospodarstev še naprej upada, je očitno, da smo zašli v sistem, kjer z zakonom, kakršen je, stvari niso lahko rešljive. Trg je prost in ni omejitev glede uvoza hrane. Po drugi strani pa veljajo znotraj EU določeni dogovori in tako hrano uvažamo iz držav, kjer so cene neprimerno dražje. V Makedoniji na primer se je paradižnik na tržnicah leta 2014 prodajal za cca 20 centov, pa iz nje uvozimo najmanj živil (glej graf na začetku poglavja prehranska varnost in samooskrba). Rešitve je zato morda bolje iskati drugod. Da ne bo zgolj kritik, bi pa izpostavil, da so se v Novem mestu na kmetijski šoli (pod Trško goro) na pobudo Evropskega kmetijskega sklada za razvoj podeželja leta 2014 izvajala brezplačna usposabljanja za kmetovanje. Objavili so več različnih programov, med drugimi tudi ekološko pridelovanje zelenjave, preusmeritev kmetije v ekološko vinogradništvo^{LL}.

Če bi želeli na obstoječih kmetijskih površinah zagotoviti dovolj hrane za vse državljane, sta na voljo recimo ti dve rešitvi:

- Sprememba prehranjevalnih navad, preklon na vegetarijanstvo, ki omogoča, da se na manjši površini prehrani več ljudi.
- Prehod na tak način kmetovanj, kjer se lahko donos živil na enoto površine precej poveča. Ena od takšnih možnosti je na primer permakultura. Žal pa slednja zahteva veliko večji vložek časa za pobiranje pridelkov in veliko več znanja o stotinah komplementarnih rastlinah, tako da bi bilo pričakovati prehod na permakulturo na državni ravni skoraj utopično. Kakšnih znanstvenih raziskav na to temo ni, gre za izkušnje, ki se predajajo po ustnem izročilu... lahko pa v razmislimo o objavi Charlesa Eisensteina^{MM}; »*Running a CSA for 300-450 people on two acres of land, he achieved yields eight times what the Department of Agriculture says is possible per square foot.*«

Če pa se odločimo ohraniti prehranjevalne navade in ne preveč povečati časa, potrebnega za pobiranje hrane, pa nam ostane enostavna rešitev: povečanje površine kmetijskih površin.

Ena od možnosti bi bila tudi bolj izpopolnjena industrijska pridelava z bolj izpopolnjeno kmetijsko tehnologijo, ki pa bi imela za posledico uničevanje ekosistema, kar je v bistvu najslabša izbira. Aplikacija zato namenoma podpira zgolj ekološke kmetije.

^{LL} <http://www.ksgrm.net/obvestila/ukrep-111>

^{MM} <http://charleseisenstein.net/permaculture-and-the-myth-of-scarcity>

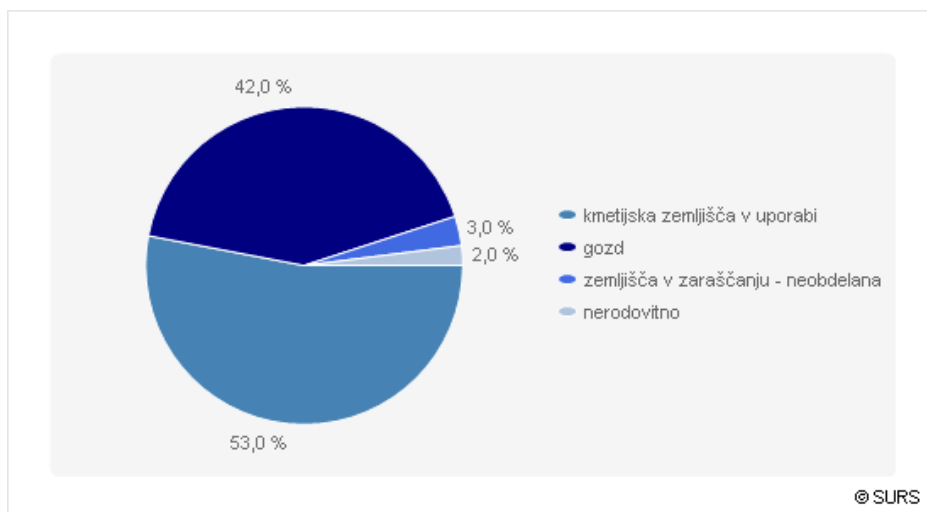
3.2 Ali lahko povečamo kmetijske površine?

V tabeli v poglavju prehranska varnost in samooskrba lahko vidimo, da je bilo pred dobrimi 40 leti v Sloveniji približno 2 x več zemljiških površin, med tem ko se je število prebivalcev povečalo le za cca 16%. Iz tega lahko sklepamo, da je potenciala za povečanje kmetijskih površin več kot dovolj.

»Slovenija z domačo pridelavo ne pokriva svojih potreb po kmetijsko-živilskih proizvodih. Vendar strokovne ocene kažejo, da Slovenija razpolaga z dovolj velikimi potenciali kmetijskih zemljišč, da prehransko varnost bistveno poveča do leta 2030.« [1d]

Perpar in Udovč [**Napaka! Vira sklicevanja ni bilo mogoče najti.**] navajata, da je glede agotavljanja prehranske varnosti na podlagi lastnih kmetijskih zemljišč Slovenija povsem na koncu lestvice držav članic Evropske unije. »Slovenija z domačo pridelavo ne pokriva svojih potreb po kmetijsko-živilskih proizvodih, stopnja samooskrbe pa je za živalske proizvode precej višja kot za rastlinske proizvode.« Dokument obenem odgovori na vprašanje v naslovu - opozarja, da je z vidika samooskrbe hrane v Sloveniji zaskrbljujoče pozidavanje in spreminjanje namembnosti kmetijskih zemljišč z najboljšim pridelovalnim potencialom. Če ta proces obrnemo, lahko povečamo kmetijske površine. Poleg tega bi se lahko kaj naredilo tudi glede zapuščenih kmetij, ki jih je v Sloveniji več tisoč.

Kakšno je stanje zemljišč? → Končni podatki za leto 2013^{NN} so prikazani na sliki dodatka 9. Predstavijo nam graf deleža zemljišč, ki jih je uporabljalo povprečno kmetijsko gospodarstvo v Sloveniji leta 2013



Slika dodatka 9: Graf deleža zemljišč (kmetijska zemljišča v uporabi, gozd, zemljišča v zaraščanju - neobdelana, nerodovitno)

^{NN} http://www.stat.si/novica_prikazi.aspx?id=6352

4 Prednosti ekološkega kmetovanja

- Kmetijsko gospodarska zbornica Slovenije je izdala brošuro, v kateri promovira ekološko kmetijstvo. Zapisali so: *»Pristopi ekološkega kmetijstva v pridelavi, reji in predelavi zmanjšujejo pritisk kmetovanja na naravo in okolje. Ekološko kmetijstvo zagotavlja visoko kakovostno in lokalno pridelano hrano. Poleg zmanjšanja pritiska na naravne vire je ekološko kmetijstvo tudi ekonomska priložnost. Trg ekoloških pridelkov in živil je hitro rastoč segment tržišča živil. Ekološko kmetijstvo zmanjša odvisnost kmetije od industrije repromateriala in lahko nudi kakovostne zaposlitve v celotni verigi.«*
- V zakonu za ohranjanje narave [14d] je navedeno, da se v naravnih rezervatih lahko prepove zasajanje monokultur. Zakaj le v naravnih rezervatih? Je na dolgi rok pomemben samo naravni rezervat? V ekološkem načinu kmetovanja se uporablja kolobarjenje.

[15d] navaja prednosti ekološkega kmetijstva, ki implicitno vplivajo na boljše življenje:

- *Ekološki kmetovalci si ne prizadevajo le za to, da bi vzdrževali zdravje, rodovitnost in naravno stanje tal. Tla lin njihovo strukturo poskušajo izboljšati z vnosom hranil in učinkovitim gospodarjenjem z vodo...*
- *Ekološko kmetijstvo dejavno prispeva k povečanju sposobnosti tal za zadrževanje vode in ohranjanje visoke kakovosti vodnih virov. Ekološki kmetijski sistemi pomagajo ohranjati ali celo izboljševati kakovost vode s pomočjo zmanjševanja količin kemikalij, ki se uporabljajo v kmetijstvu, in ki lahko pronicaajo v jezera, reke, potoke in v druga vodna telesa. Ekološko kmetijstvo omejuje rabo sintetičnih gnojil in pesticidov, pospeševalcev rasti in antibiotikov za živali, kar zmanjšuje tveganje, da bi te snovi zašle v vodna telesa. Zmanjšana je tudi nevarnost eutrofikacije; pri tej prekomerno razmnoževanje alg, ki ga povzroča spiranje hranil v ta vodna telesa, zmanjšuje vsebnost kisika in ogroža zdravje vodnih rastlin in živali.*
- *Nekatere študije so celo pokazale, da ekološko kmetijstvo proizvaja manj izpustov CO₂, zato bi lahko pomembno prispevalo k zmanjševanju učinkov podnebnih sprememb. Dejavniki, ki lahko prispevajo k temu, so:*
 - *zmanjšana raba energije s pomočjo recikliranja odpadkov in stranskih proizvodov namesto uporabe sintetičnih gnojil, izdelanih s pomočjo energetske intenzivnih postopkov,*
 - *ohranjanje več prvotne vegetacije in več vegetacije na splošno;*
 - *skladiščenje ogljika v organski snovi tal, saj skušajo ekološki kmetovalci povečati vsebnost humusa v tleh.*
- *Raziskovalni inštitut za ekološko kmetijstvo (FiBL) (en) v svoji študiji ugotavlja, da so emisije toplogrednih plinov na hektar v ekoloških sistemih za 32 % nižje kot v sistemih z mineralnimi gnojili in za 35–37 % nižje kot v konvencionalnih sistemih na osnovi hlevskega gnoja. Po tej študiji je to povezano z dejstvom, da ekološko kmetijstvo v povprečju vrne v tla 12–15 % več ogljikovega dioksida kot sistemi z mineralnimi gnojili, in sicer s pomočjo spodbujanja rodovitnosti tal in vsebnosti humusa.*
- *Ekološko kmetijstvo skrbi za biotsko raznovrstnost in si prizadeva za njeno izboljšanje, kjer je to mogoče. Ekološki kmetje plevel zatirajo z rokami in ne s herbicidi. To pomaga ohranjati biotsko raznovrstnost tako na površju kot pod njim. Ko v ekološkem kmetijstvu*

uporabimo izraz biotska raznovrstnost, to ne pomeni le večjega števila rastlin in živali, ampak tudi to, da več rastlin in živali, značilnih za določeno območje, raste ali živi kar najbolj naravno. Poseben poudarek je namenjen tudi ohranjanju ogroženih živalskih in rastlinskih vrst.

- Nekatere študije so ugotovile, da ekološko kmetijstvo povzroča povečanje števila koristnih organizmov, ki živijo v tleh, kar pomaga zagotavljati zdravje posevkov in živali.
- Sama narava ekološkega kmetijstva pomeni, da vodi k ustvarjanju delovnih mest, številčnejšemu podeželskemu prebivalstvu in k obilju na podeželju. Recesija je. Hočete več brezposelnih, ki se bodo preživljali na račun davkoplačevalcev, ali bi morda radi več zaposlitvenih mest? Študija britanske organizacije Soil Association iz leta 2006 z naslovom Poklici v ekološkem kmetijstvu (en) je ugotovila, da ekološko kmetijstvo v Veliki Britaniji zagotavlja 32 % delovnih mest na kmetijo več kot primerljive konvencionalne kmetije, ekološki kmetje v Veliki Britaniji pa so v povprečju tudi sedem let mlajši od svojih konvencionalnih kolegov.
- Ekološko kmetijstvo ustvarja privlačnejšo naravno krajino, tako da je osredinjeno na:
 - zasajanje živih mej in ohranjanje travnikov;
 - ohranjanje domače flore in favne;
 - varovanje in izboljševanje vodnih virov in tal;
 - uporabo avtohtonih vrst rastlin in živali ter lokalnih virov.
- Znanstveniki niso oblikovali enotnega sklepa glede tega, ali so pridelki v ekološkem kmetijstvu nižji od tistih v drugih kmetijskih sistemih. Izvedenih je bilo več študij, ki kažejo na obe strani zgodbe. Ena izmed študij, ki je ugotovila primerljivo raven pridelkov pri obeh sistemih, je pogosto navajano poročilo univerze Cornell (en) v ZDA o kmetijskem poskusu Inštituta Rodale. V okviru tega poskusa so 22 letih z ekološkim kmetovanjem pridelali enake donose pšenice in soje kot konvencionalne kmetije, toda za to so porabili 30 % manj energije, manj vode in nobenih sintetičnih pesticidov.

Literatura dodatkov

- [1d] Uredba Sveta (ES) št. 834/2007 o ekološki pridelavi in označevanju ekoloških proizvodov. Dostopno na:
http://www.arhiv.mkgp.gov.si/si/delovna_podrocja/organi_v_sestavi/inspektorat_rs_za_kmetijstvo_gozdarstvo_hrano/inspekcija_za_kontrolo_krme/zakonodaja/ (25.9.2014)
- [2d] Statistični urad RS: Splošni pregled kmetijskih gospodarstev. Dostopno na:
http://pxweb.stat.si/pxweb/Database/Okolje/15_kmetijstvo_ribistvo/03_kmetijska_gospod/00_15560_splosno/00_15560_splosno.asp (25.9.2014)
- [3d] Ministrstvo za kmetijstvo in okolje: Analiza stanja ekološkega kmetovanja. Dostopno na:
http://www.mko.gov.si/si/delovna_podrocja/kmetijstvo/ekolosko_kmetovanje/analiza_stanja_ekoloskega_kmetovanja/ (25.9.2014)
- [4d] http://www.stat.si/doc/pub/kmetijstvo_EU.pdf (25.9.2014)
- [5d] dr. Dušan Plut: Prehranska varnost sveta in Slovenije, Oddelek za geografijo, Filozofska fakulteta Univerze v Ljubljani. Dostopno na
<http://www.dlib.si/> (25.9.2014)
- [6d] Statistični letopis 1953, 1970, 2012, Kmetijstvo (in ribištvo). Dostopno na:
http://www.stat.si/publikacije/pub_letopis_prva.asp (25.9.2014)
- [7d] Statistični urad: Popis kmetijstva 2010. Dostopno na:
<http://www.stat.si/PopisKmetijstva2010/> (25.9.2014)
- [8d] Statistični urad: Struktura kmetijskih gospodarstev, podrobni podatki, Slovenija in statistične regije, 2013 - končni podatki. Dostopna na:
http://www.stat.si/novica_prikazi.aspx?id=6352 (25.9.2014)
- [9d] Špela Gale, Statistični urad: Statistične informacije, kmetijstvo in ribištvo. Dostopno na:
<http://www.stat.si/doc/statinf/15-si-277-1201.pdf> (25.9.2014)
- [10d] STELE, Aleš: O kmetijstvu doma in drugje po EU [Elektronski vir] / [avtorja Aleš Stele in Irena Žaucer]. - El. knjiga. - Ljubljana : Statistični urad Republike Slovenije, 2013. - (Zbirka Brošure / Statistični urad Republike Slovenije). Dostopno na:
http://www.stat.si/doc/pub/kmetijstvo_EU.pdf (25.9.2014)
- [11d] Statistični urad: Kmetijska gospodarstva po velikostnih razredih kmetijskih zemljišč v uporabi (KZU). Dostopno na:
https://www.stat.si/letopis/2013/16_13/16-02-13.html (25.9.2014)
- [12d] Suhadolc M., Sušnik A., Lobnik F., Kajfež Bogataj L., Gregorič G., Bergant K., 2010, Izzivi Slovenije na področju suš in degradacije tal, str. 67. ARSO, Biotehniška fakulteta, Ljubljana. Dostopno na:
http://meteo.arso.gov.si/uploads/probase/www/agromet/product/document/sl/IZZIVI_Slovenije_na_podrocju_sus_in_degradacije_tal.pdf (25.9.2014)

- [13d] Perpar Anton, Udovč Andrej, 2010: Realni potenciali za lokalno oskrbo s hrano v Sloveniji. Dela 34, str. 187–199. Dostopno na: <http://www.dlib.si/> (25.9.2014)
- [14d] Uradni list RS, št. 56/1999: Zakon o ohranjanju narave (ZON), Stran 7146. Dostopno na: <https://www.uradni-list.si/1/content?id=20635> (25.9.2014)
- [15d] http://ec.europa.eu/agriculture/organic/what-organic_sl (25.9.2014 ni dostopno; dobra alternativa: http://www.zpm-mb.si/attachments/sl/271/Ali_so_ekoloska_zivila_res_boljsa.pdf)